

UNIVERSITÀ DEGLI STUDI DI PISA

Facoltà di Ingegneria

Corso di Laurea in Ingegneria Aerospaziale

Tesi di Laurea

**Sviluppo di metodi per il progetto
ottimizzato di strutture alari**



Relatori:

Prof. Ing. Attilio Salvetti

Dott. Ing. Daniele Fanteria

Candidato:

Roberto Zuddas

Anno accademico 2008-2009

Nelle fasi iniziali del progetto strutturale, oltre all'esperienza e la creatività, occorrono delle informazioni utili per valutare un gran numero di soluzioni nel minor tempo possibile. In quest'ottica nel presente lavoro di tesi si delinea una procedura di progetto ottimizzato di strutture alari e viene sviluppato uno strumento di calcolo che la realizza. Detto strumento è ottenuto mediante *software* di varia natura: si usa *modeFrontier* in qualità di ambiente di processo ed ottimizzazione, *MD PATRAN* come pre e post processore grafico e *NASTRAN* come solutore *FEM*; in più vengono sviluppati codici di servizio in linguaggio *MATLAB*.

Lo strumento che ne risulta permette di impostare le variabili ed i parametri di progetto (dimensioni geometriche della struttura e dati del materiale), costruire un modello tridimensionale coerente con i dati forniti e sottoporlo ad analisi. *ModeFrontier* gestisce il ciclo di analisi in modo completamente automatico e permette ottimizzazioni vincolate multiobiettivo mediante diversi algoritmi (genetici, gradientali, teoria dei giochi). Vengono presentati esempi di ottimizzazioni condotte con algoritmi genetici che generano, a partire da una popolazione iniziale di configurazioni ammissibili, nuove soluzioni in base agli obiettivi ed ai vincoli imposti.

Gli esempi studiati hanno mostrato che lo strumento di analisi e ottimizzazione permette di valutare correttamente gli stati di tensione e deformazione ai quali lavora la struttura. Il programma è ancora in fase di sviluppo, in particolare utilizza la sequenza di soluzione lineare del *NASTRAN* e presuppone un costituente isotropo; l'obiettivo per il futuro è di estendere il procedimento ai materiali ortotropi e di aggiungere la possibilità di svolgere analisi non lineari.

Sommario	iii
1 Introduzione	1
1.1 Obiettivi del presente lavoro	1
2 Definizione del modello ad elementi finiti	3
2.1 Geometria esterna di riferimento	5
2.2 Il modello <i>FE</i>	7
2.3 Vincoli e carichi sul modello	13
2.4 Il modello completo	17
3 Definizione dei carichi e degli ammissibili	21
3.1 Introduzione	21
3.2 Classificazione dei carichi	22
3.3 Carichi in volo	24
3.3.1 Condizioni simmetriche	24
3.3.2 Condizioni non simmetriche	27
3.4 Carichi a terra	29
3.4.1 Atterraggio	29
3.4.2 Condizioni al suolo	29
3.5 Scelta delle condizioni di volo e calcolo dei carichi	30
3.5.1 Calcolo delle risultanti aerodinamiche	32
3.5.2 Calcolo del momento aerodinamico	33
3.5.3 Calcolo della risultante di massa strutturale e combustibile	34
3.5.4 Calcolo dei momenti di trasporto	35

3.6	Scelta delle tensioni dimensionanti	36
3.7	Scelta degli ammissibili	37
3.7.1	Margini di sicurezza e controlli	37
3.7.2	Caratteristiche del materiale	37
3.7.3	Ammissibili ad instabilità	37
3.7.4	Ammissibili a fatica	41
4	Ciclo di ottimizzazione	45
4.1	Introduzione al <i>modeFrontier</i>	45
4.1.1	Flusso logico	46
4.1.2	Flusso dati	47
4.2	Schema logico del ciclo	48
4.3	Scelte di vincoli e funzioni obiettivo	54
5	Esempi di progetto ottimizzato di strutture	57
5.1	Ottimizzazione del passo dei correnti	57
5.2	Confronto: soluzione più leggera e più rigida	64
5.3	Ottimizzazione preliminare delle centine	77
6	Sviluppi futuri	87
6.1	Analisi non lineare	88
6.1.1	Introduzione	88
6.1.2	Algoritmi di soluzione	89
6.1.3	Applicazione: analisi dei longheroni	94
6.2	I materiali compositi	97
6.2.1	Introduzione	97
6.2.2	Teoria dei laminati	97
6.2.3	Alcune indicazioni sull'ottimizzazione di strutture in composito al <i>modeFrontier</i>	101
7	Conclusioni	105
	Appendici	109
A	Breve guida all' uso del software	109
A.1	Principali estensioni dei file	110
A.2	Il file .prj di <i>modeFrontier</i>	111
A.3	Il codice Matlab	115
A.4	Il <i>Patran command language</i>	117

B	Il processo di creazione del modello	121
B.1	Creazione della geometria	126
B.2	Generazione della <i>mesh</i>	129
B.3	Assegnazione dei vincoli e dei carichi	136
B.4	Attribuzione delle proprietà e creazione dell' <i>input</i> Nastran	138
C	Il processo di verifica delle dimensioni	141
D	La <i>card</i> NLPARM	143
	Bibliografia	147
	Elenco delle figure	148
	Elenco delle tabelle	152

1.1 Obiettivi del presente lavoro

Nelle prime fasi del processo di progettazione strutturale sono necessarie informazioni di base che guidino le decisioni sulla configurazione da adottare per le componenti da realizzare; questo richiede esperienza e creatività, ma servono anche e sempre di più metodi analitici e numerici che aiutino il progettista nelle sue scelte.

Preliminarmente si cerca di valutare un gran numero di soluzioni nel minor tempo possibile, dunque uno strumento in grado di fornire con semplicità risposte rapide a domande sull'architettura migliore per la struttura può avere un grande valore poiché permette di operare decisioni critiche, suffragate da simulazioni con modelli affidabili.

Nel presente lavoro di tesi si cercherà di delineare una procedura automatica per il progetto ottimizzato del cassone alare di un velivolo da trasporto civile, che deve essere economica ed eseguibile in tempi relativamente brevi, in quanto concepita come ausilio nella scelta delle principali dimensioni geometriche nelle prime fasi del progetto di un aeromobile, nelle quali occorra valutare un gran numero di configurazioni strutturali diverse.

La procedura concepita inizia con la definizione, sulla base della geometria esterna, assegnata, del cassone alare, rappresentato da un opportuno modello ad elementi finiti che sia un compromesso tra le contrastanti esigenze di accuratezza e semplicità. La scelta ricade su una *mesh* piuttosto grossolana, fatta di elementi monodimensionali per gli irrigidimenti e le flange dei longheroni e bidimensionali per i pannelli, le anime e le centine. Un tale modello si presta ad analisi preliminari che hanno l'obiettivo di mettere in evidenza la ripartizione dei carichi all'interno della struttura. Le linee guida seguite per la concezione e costruzione del modello della struttura portante vengono illustrate nel capitolo 2.

Una volta realizzato il modello del cassone, occorre evidenziare quali siano i carichi che possono gravare sulla struttura nel corso della sua vita operativa. Fatta una dovuta premessa riguardante le richieste da parte delle normative vigenti in campo aeronautico per quanto concerne le condizioni di volo che un aeromobile può riscontrare durante la sua missione, si illustreranno in seguito i casi di carico scelti per l'analisi ed i valori massimi consentiti delle sollecitazioni. Quest'argomento sarà sviluppato nel capitolo 3 del presente lavoro.

Per eseguire il compito prefissato, si ricorrerà in modo congiunto ad un solutore di modelli ad elementi finiti per analisi strutturali, *MD NASTRAN*, ed il relativo pre e post-processore, *MD Patran*, della *MSCSoftware*, un codice per il calcolo numerico, *Matlab* della *Mathworks* ed un ambiente di ottimizzazione ed integrazione di processi, *modeFrontier* della *ESTECO*.

Il processo logico di ottimizzazione della struttura è l'oggetto del capitolo 4, nel quale, fatta una dovuta premessa sull'ambiente di integrazione di cui ci si è serviti per il presente lavoro, si indicheranno dettagliatamente le assunzioni e le scelte adottate per sviluppare il codice di simulazione. Nel capitolo 5 saranno illustrati i risultati, ottenuti per il modello di cassone alare scelto come riferimento, di alcune simulazioni esemplificative svolte nell'ultima fase del presente lavoro.

Il risultato finale è uno strumento versatile, che consente rapidi studi di sensibilità così come analisi approfondite alla fine delle quali si ottengono soluzioni che, nei limiti definiti, sono di ottimo, vale a dire di minimo peso ed efficienti. In particolare, la possibilità di sfruttare le potenzialità degli algoritmi risolutori di nuova concezione e dei più recenti mezzi di calcolo elettronico permette di affrontare un problema molto complesso e ricco di variabili con uno spirito nuovo.

Si è scelto un modello che non fosse eccessivamente complesso e si è adottato il metodo delle analisi lineari per la soluzione strutturale supponendo un materiale costituente di tipo metallico. Queste premesse sono state utili per sviluppare lo strumento in un ambito più facilmente gestibile per non doversi preoccupare di complicazioni sulla natura del materiale o sul tipo di analisi e concentrarsi soltanto sulla necessità di mettere a punto un procedimento funzionale e modulare. Nel corso del capitolo 6 si cerca di tracciare una possibile strada da seguire nel caso si voglia approfondire l'argomento al fine di ottenere soluzioni mediante analisi non lineari o applicate a strutture costituite da materiali innovativi quali i compositi in fibra di carbonio e matrice epossidica, che hanno un crescente impiego nel settore aeronautico.

CAPITOLO 2

DEFINIZIONE DEL MODELLO AD ELEMENTI FINITI

Nell'intento di illustrare la procedura automatica sviluppata per il progetto di configurazioni strutturali dell'ala, si comincia mostrando il modello di cassone alare di riferimento e le approssimazioni e semplificazioni introdotte.

Le linee guida scelte per la creazione dei modelli per l'ottimizzazione sono:

- *Semplicità*: il modello deve essere costituito dal minor numero di elementi possibile, in modo da garantire tempi di calcolo ridotti, ma comunque tale da fornire un risultato attendibile
- *Parametrizzazione*: le principali caratteristiche geometriche non ancora assegnate devono poter essere modificate dal *software* senza intervento dell'operatore

Per questo si è utilizzato l'ambiente **MATLAB** per la scrittura del *file* di sessione del preprocessore, **PATRAN**. Questo contiene la sequenza di operazioni da eseguire per la creazione della geometria dell'ala e del cassone e successivamente per l'analisi strutturale vera e propria, svolta con il codice ad elementi finiti **MD NASTRAN**. Il processo di ottimizzazione e di elaborazione dei risultati è stato gestito con **modeFRONTIER**.

L'utilità di gestire il processo di generazione delle configurazioni con **MATLAB** risiede nel fatto che in questo modo è possibile modificare a piacimento i parametri geometrici della struttura, scrivere una sessione con valori modificati e generare un modello differente senza dover ripetere manualmente le operazioni, con evidenti vantaggi in termini di praticità.

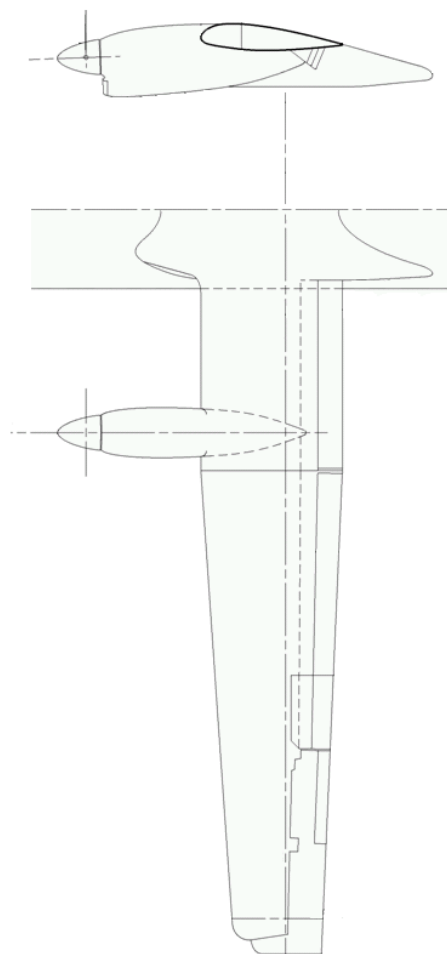


Figura 2.1: Ala del velivolo di riferimento

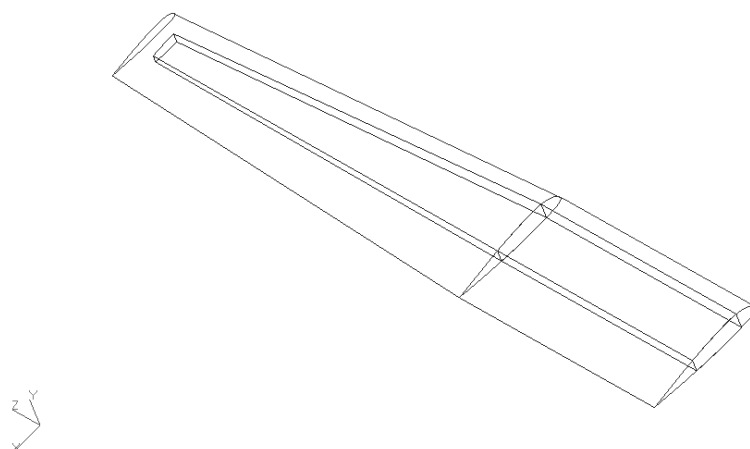


Figura 2.2: Superficie e cassone alare di riferimento

2.1 Geometria esterna di riferimento

Nella procedura delineata si parte da una forma in pianta dell'ala, assegnata come dato del problema, dalla quale si realizza il modello del cassone alare. In particolare, per questo lavoro si è scelta la superficie portante di un generico velivolo da trasporto civile regionale ad ala alta, bimotore a breve o media tratta, riportata in figura 2.1 e 2.2.

Apertura	27.05 m
Superficie	61 m^2
Allungamento	12
Distanza dalla radice al <i>kink</i>	4.75 m
Corda alla radice	2.57 m
Corda all'estremità	1.59 m
Rapporto di rastremazione	0.61
Freccia al bordo d'attacco	3.9°
Freccia al bordo d'uscita	-2.5°
Spessore percentuale alla radice ed al <i>kink</i>	0.18
Spessore percentuale al <i>tip</i>	0.13

Tabella 2.1: Principali dimensioni geometriche dell'ala

Distanza dalla radice all'estremità	13.12 m
Corda alla radice	1.13 m
Corda al <i>tip</i>	0.53 m
Passo centine alla radice	0.3025 m
Distanza del motore dal <i>kink</i>	0.35 m
Distanza orizzontale del baricentro del motore dal longherone	0.9 m
Distanza verticale del baricentro del motore dal longherone	0.35 m

Tabella 2.2: Principali dimensioni geometriche del cassone

L'ala ha una parte *inboard* dritta ed una *outboard* rastremata in corda ed in altezza. Il cassone è cilindrico fino al *kink* e prosegue all'estremità con angolo di freccia positivo sul longherone anteriore e nullo sul longherone posteriore. I motori, di tipo turboelica, sono montati nella parte dritta dell'ala in prossimità della penultima centina prima del *kink*. L'attacco in fusoliera viene realizzato mediante quattro puntelli che collegano i longheroni con due ordinate in corrispondenza della quarta centina, e quattro staffe che vincolano a traslazione longitudinale la struttura in corrispondenza della seconda centina.

Le principali dimensioni geometriche dell'ala e del cassone sono riportate in tabella 2.1 e 2.2 rispettivamente.

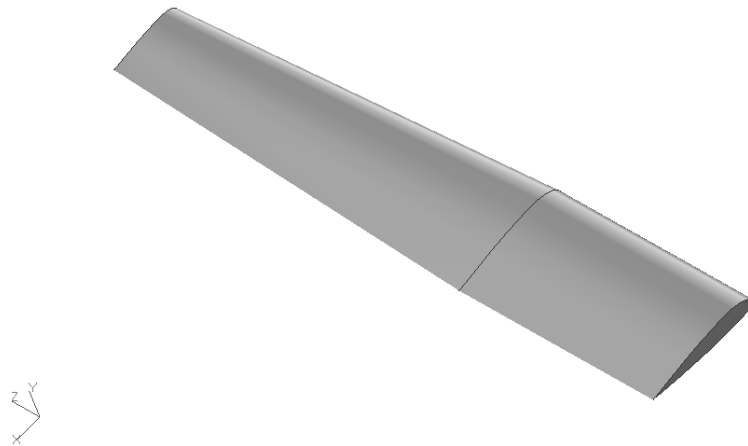


Figura 2.3: Modello **Patran** della superficie alare

2.2 Il modello *FE*

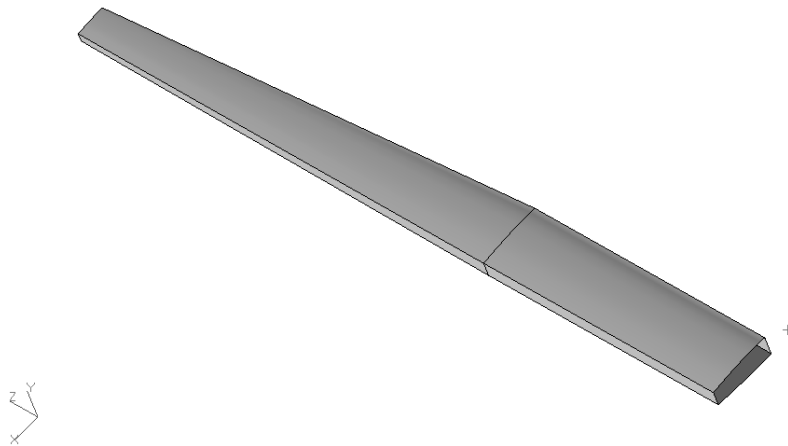
A partire dai dati sui profili alari¹, si tracciano le tre curve alla radice, al *kink* ed all'estremità, dalle quali si ricostruisce la superficie bagnata dell'ala. Da questa si ottiene il contorno del cassone operando dei tagli verticali in corrispondenza dei longheroni e dell'estremità dello stesso. In figura 2.3 è riportato il modello della superficie esterna dell'ala (nel quale si distinguono le curve dei profili aerodinamici) mentre in figura 2.4 viene rappresentato il cassone alare.

I principali parametri geometrici interessati dal ciclo di ottimizzazione sono la spaziatura delle centine nei tratti diritto e rastremato, il passo dei correnti nei pannelli dorsale e ventrale, tutti gli spessori, le caratteristiche delle sezioni dei correnti e delle flange dei longheroni (area, posizione del baricentro, momenti di inerzia). Sono stati inoltre definiti alcuni fattori di controllo della *mesh* quali il numero di elementi nei pannelli, nei longheroni e nelle centine (si veda la tabella 4.1 per la lista completa delle variabili introdotte nel processo di ottimizzazione).

I pannelli, le centine e le anime dei longheroni vengono modellati mediante elementi *shell* con spessore; i correnti e le flange dei longheroni con elementi *bar* dei quali si sono definiti area, momenti di inerzia e distanza (*offset*) del baricentro della sezione dal piano medio dei pannelli.

Si è scelto di rappresentare i setti ed i longheroni mediante elementi tipo piastra. Per gli elementi monodimensionali, al fine di calcolarne le caratteristiche geometriche, si è posto

¹Tratti da http://www.ae.uiuc.edu/m-selig/ads/coord_database.html ed opportunamente scalati

Figura 2.4: Modello **Patran** del cassone alare

che gli irrigidimenti sui pannelli avessero sezione a **zeta** e le flange dei longheroni a **t**. In figura 2.5 è rappresentata una tipica sezione del cassone, semplificata per la simulazione.

L'interruzione degli irrigidimenti longitudinali viene gestita come mostrato in figura 2.6: la trama di elementi finiti del corrente termina in corrispondenza della centina precedente al *run out* ma, per esigenze di coerenza dei nodi, vengono creati delle parti di forma trapezia e triangolare al fine di evitare che un elemento con tre nodi su un lato sia adiacente ad uno con due soli nodi sul lato contiguo, come accade in figura 2.7.

Le centine (figura 2.8) sono costituite da due strisce di transizione in corrispondenza dei pannelli e da una parte centrale di elementi regolari. Queste zone sono necessarie per mantenere la corrispondenza fra i nodi nel lato superiore ed inferiore della centina a causa della diversa spaziatura tra irrigidimenti sul rivestimento dorsale e ventrale. In generale il codice scritto realizza un modello con una spaziatura delle centine dovuta al valore in ingresso per la zona *inboard* ed *outboard*; fanno eccezione le prime quattro centine alla radice e le ultime due prima del *kink*, le quali hanno passo fisso. Questa scelta è dovuta al fatto che al variare degli *input* potrebbe modificarsi la posizione ed il numero delle centine, ma i setti in corrispondenza dei quali avviene l'attacco in fusoliera - il secondo ed il quarto - e quello sul quale sarà il motore - l'ultimo prima del *kink* - devono rimanere nella stessa posizione in apertura.

La qualità della *mesh* sulla centina è tanto migliore quanto più sono vicini i valori delle spaziature dei correnti dorsali e ventrali, come in figura 2.9. Se invece questi valori dovessero differire sensibilmente, aumenterebbe il numero di elementi poco regolari, come in figura 2.10. Nel codice è stata introdotta una variabile, *rib_param*, modificabile dall'utente,

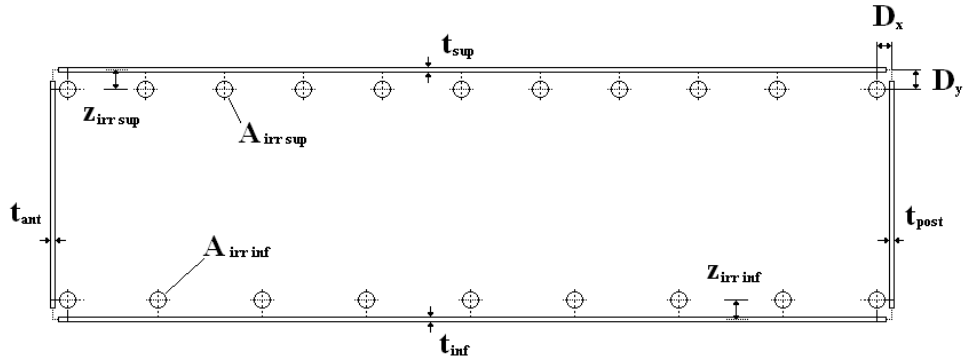


Figura 2.5: Sezione tipica del cassone

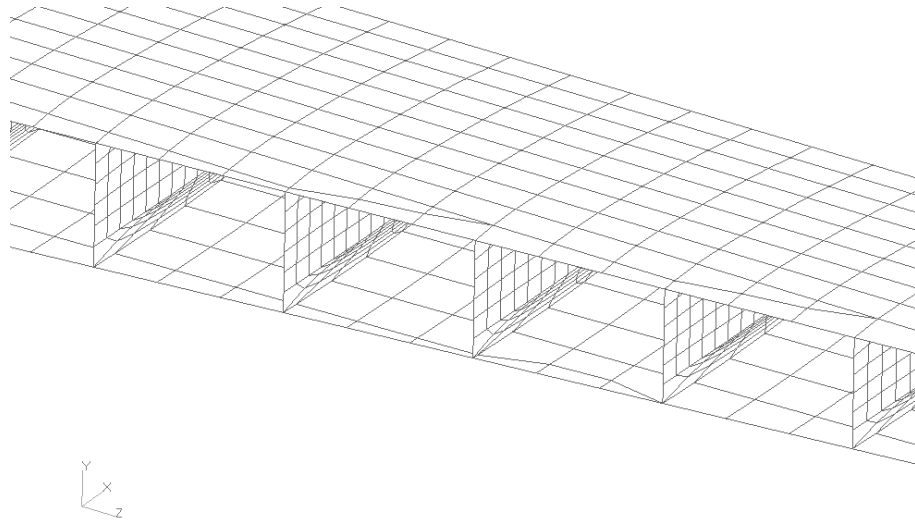


Figura 2.6: *Run out* dei correnti : dettaglio

che permette di definire quanti elementi creare nella parte centrale della centina, secondo la relazione seguente, opportunamente arrotondata

$$N_{nodi,centrale} = rib_param \cdot \frac{N_{nodi,dorso} + N_{nodi,ventre}}{2} \quad .$$

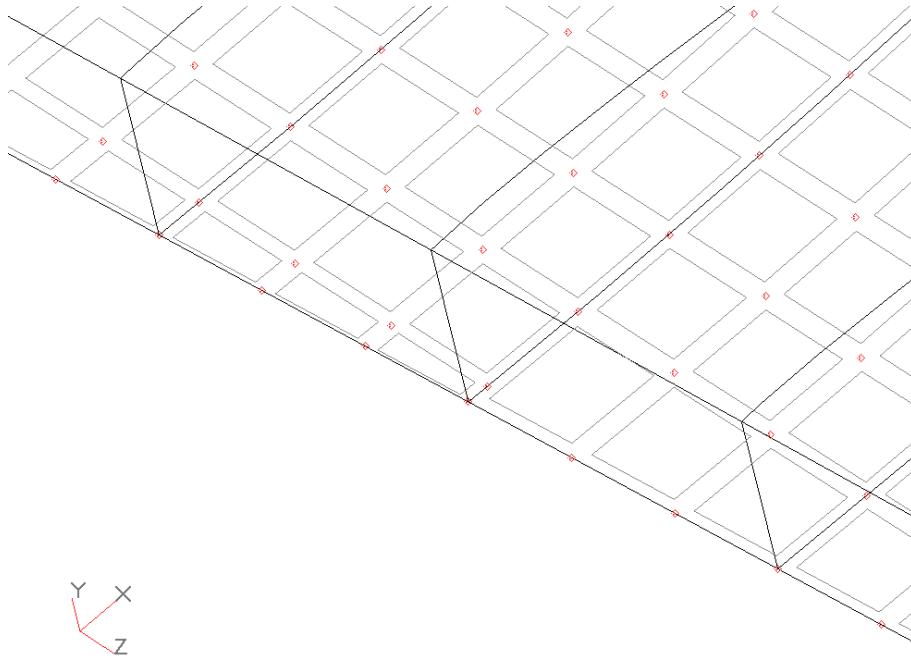


Figura 2.7: Esempio di errore nella modellazione del *run out*

A titolo di esempio, si riporta, in figura 2.11, un setto realizzato avendo posto questo parametro pari a 0.5 (i valori relativi alle figure precedenti sono rispettivamente 1 e 2).

Per determinati valori della spaziatura dei setti è possibile che l'ultima baia, interna o esterna, sia eccessivamente lunga; in questi casi il codice provvede ad inserire una ulteriore centina per riempire lo spazio.

Tutto il cassone è rastremato negli spessori e nelle altezze dei correnti. Il codice riceve in *input* i valori relativi a tre stazioni (radice, *kink*, estremità) ed opera una interpolazione per le baie intermedie supponendo un andamento lineare della rastremazione lungo l'apertura.

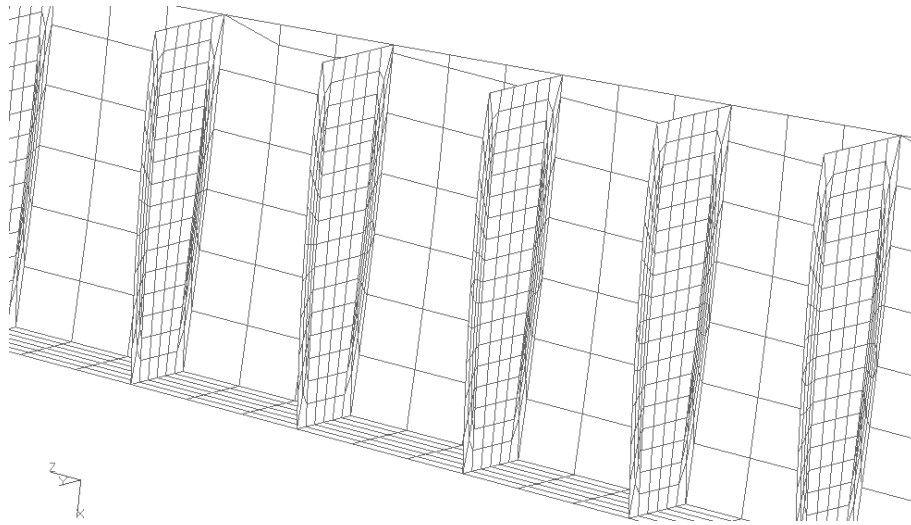


Figura 2.8: Le centine nel modello *FE*

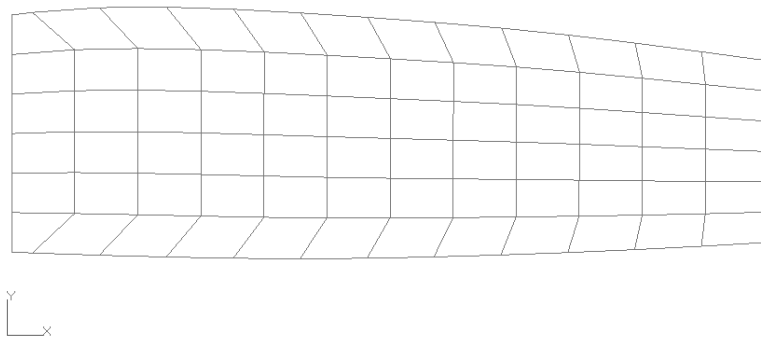


Figura 2.9: *Mesh* delle centine: caso in cui le spaziature dei correnti sono uguali

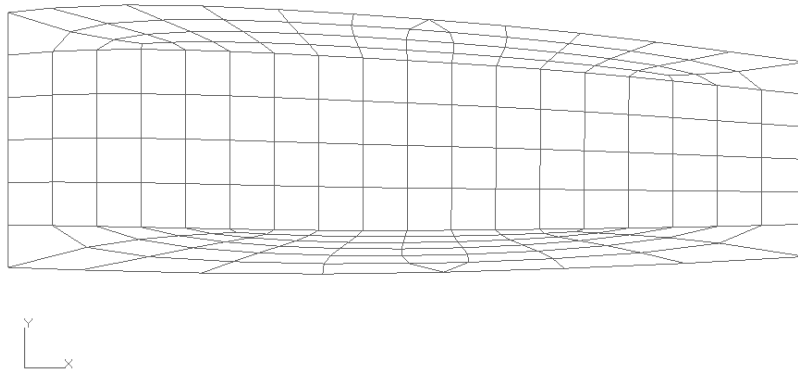


Figura 2.10: *Mesh* delle centine, caso con spaziature molto diverse tra dorso e ventre

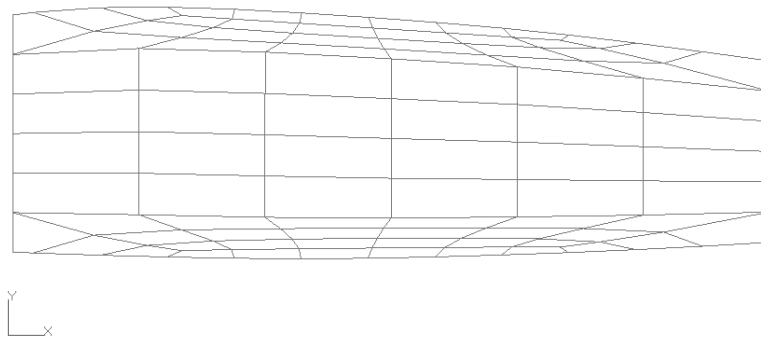


Figura 2.11: *Mesh* delle centine, $rib_param = 0.5$

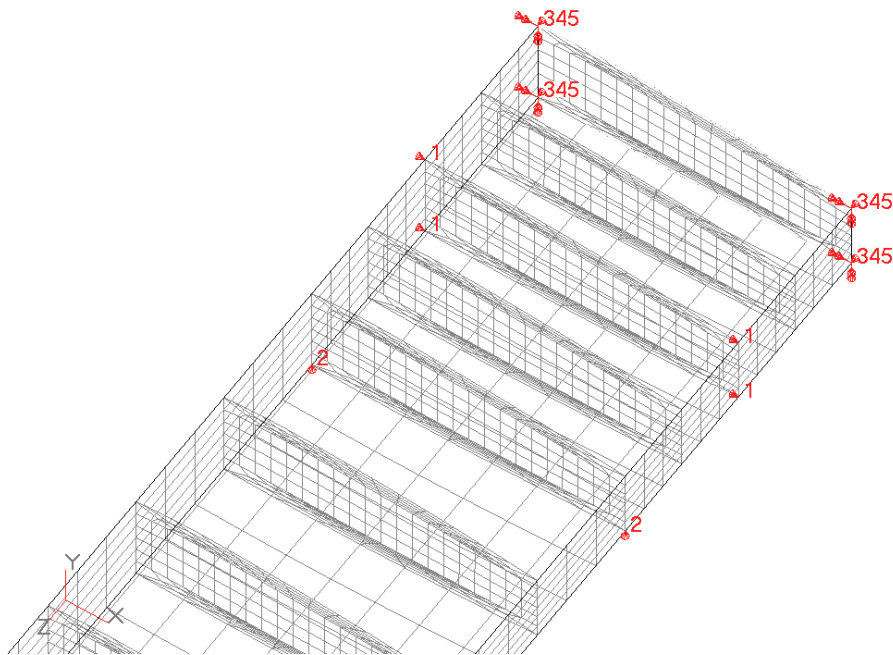


Figura 2.12: Vincoli sul modello

2.3 Vincoli e carichi sul modello

Il modello agli elementi finiti rappresenta una semiala del velivolo. La condizione di vincolo alla radice è un doppio pendolo tridimensionale sul piano di simmetria in corrispondenza della prima centina, due appoggi verticali nell'incontro tra longheroni e quarto setto e due vincoli alle traslazioni orizzontali. Le condizioni appena descritte sono riportate in figura 2.12.

Le forze che gravano sul cassone vengono introdotte nelle centine mediante *MPC RBE3* (figura 2.13) nei quali il nodo indipendente - corrispondente alla traccia dell'asse di riferimento nel piano della centina - si trova nel centro della superficie del setto ed i nodi indipendenti si trovano sul perimetro, come mostrato in figura 2.14. Si è scelto questo tipo di vincolo per la necessità di distribuire uniformemente un carico concentrato equivalente alle caratteristiche della sollecitazione nel cassone; ogni centina del modello prende la forza concentrata applicata sulla traccia dell'asse e la ripartisce equamente su tutto il perimetro. Un ulteriore punto di introduzione dei carichi è l'attacco del motore, modellato in corrispondenza della penultima centina *inboard* nello stesso modo ed avente il nodo dipendente che rappresenta il baricentro del gruppo motore e gondola, in posizione avanzata e verso il basso rispetto alla flangia inferiore del longherone anteriore (tabella 2.2 e figura 2.15). Si tratta questo di uno schema piuttosto semplificato della struttura portante del motore, che si potrà rappresentare

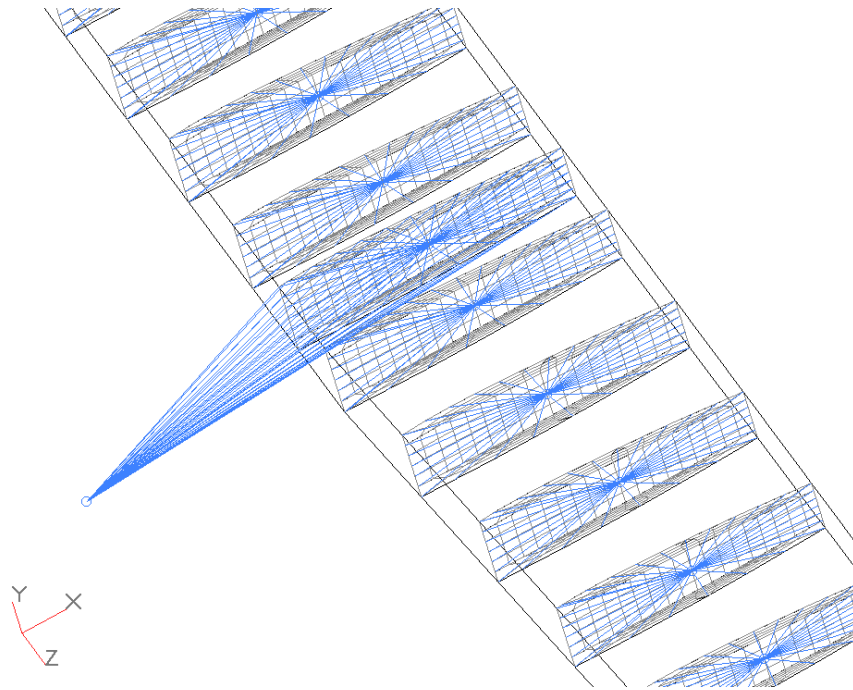


Figura 2.13: Introduzione dei carichi aerodinamici ed inerziali: *MPC RBE3*

in seconda approssimazione con una travatura reticolare.

In figura 2.16 è riportato uno schema di come sono state calcolate le forze agenti sugli *MPC*. La risultante delle forze e dei momenti aerodinamici (in azzurro) agisce per ipotesi al 25% di corda alare, la massa della struttura (in nero) si assume concentrata a metà della corda alare e la massa del combustibile (in rosso) a metà della corda del cassone. Per ogni condizione di volo che verrà analizzata, è possibile stabilire l'entità di queste forze e momenti.

Infatti, note che siano le condizioni di peso complessivo, quota, velocità e fattore di carico della condizione in esame è possibile determinare la forza e la coppia aerodinamica, il peso strutturale e del combustibile setto per setto. Si rimanda al paragrafo 3.5 per ulteriori dettagli sulle condizioni di volo scelte per la simulazione e sulle approssimazioni usate per il calcolo.

Per comodità, si carica il modello con forze e momenti **unitari**, agenti sui nodi indipendenti degli *RBE3* in figura 2.17. Una volta scelte le condizioni dimensionanti e calcolata la risultante delle forze e dei momenti baia per baia, ciascuna di queste viene rappresentata da un *load case* di **NASTRAN** composto da tutti i carichi unitari moltiplicati per il valore effettivo della sollecitazione.

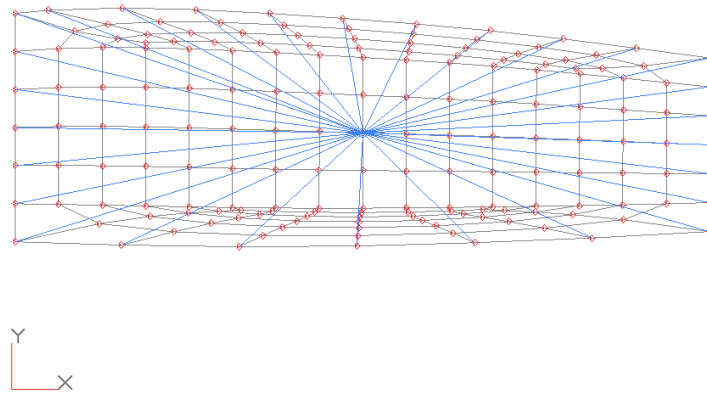


Figura 2.14: Dettaglio di un *RBE3*

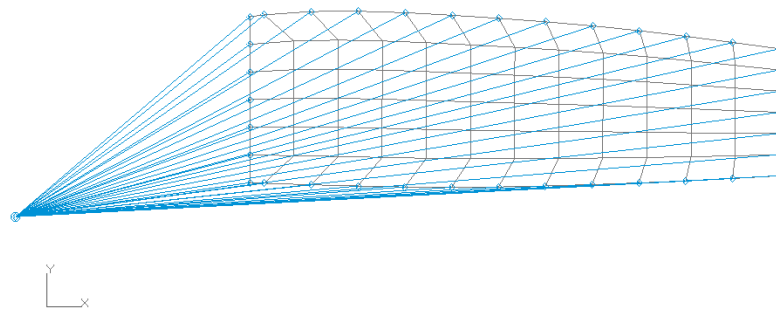


Figura 2.15: *MPC* che rappresenta l'attacco del motore

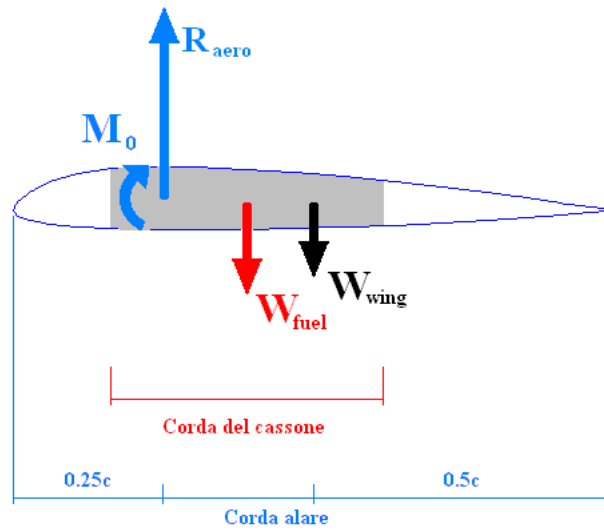


Figura 2.16: Forze agenti sul profilo: risultante verticale e momento aerodinamici (*azzurro*), distribuzione di massa strutturale (*nero*) e distribuzione di massa del combustibile (*rosso*)

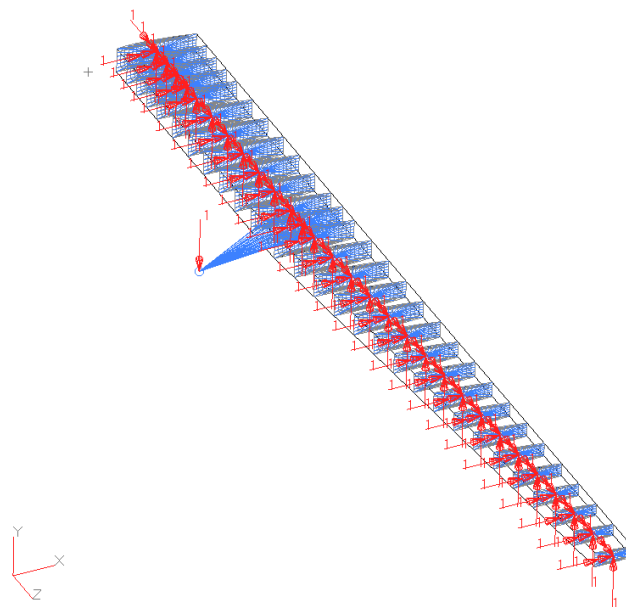


Figura 2.17: Carichi unitari agenti sulle centine del modello

2.4 Il modello completo

Seguono le immagini del modello ottenuto dall'ala di partenza, pronto per essere analizzato. In figura 2.18 si è usata la rappresentazione in *wireframe*, nella quale si può apprezzare la differenza di spaziature tra centine alla radice, *inboard*, in zona motore ed *outboard*. In figura 2.19 vi è il modello visualizzato in modalità *hidden line*, con la quale si vedono bene le interruzioni dei correnti sul pannello superiore. In figura 2.20, infine, è possibile vedere il modello con le ombreggiature (*shaded*), probabilmente la più immediata fra le tre.

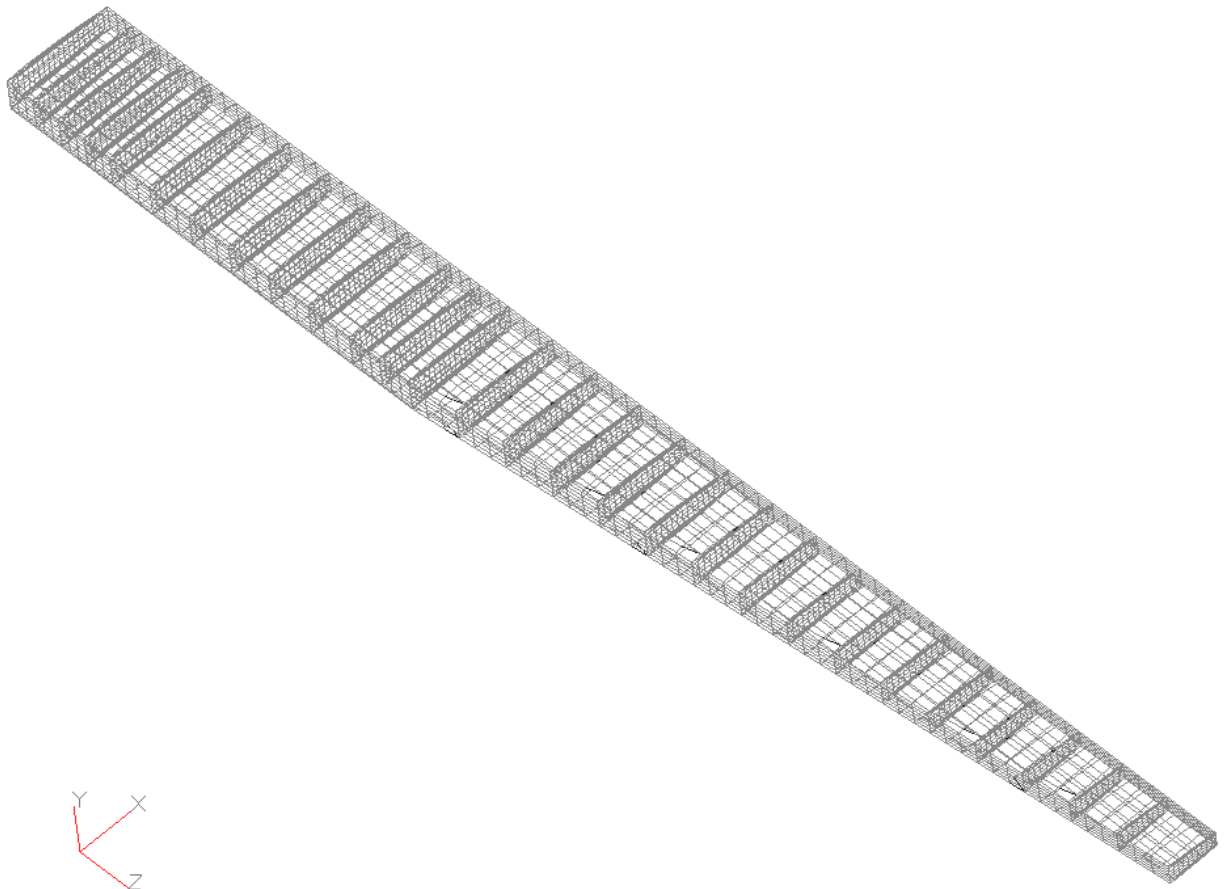


Figura 2.18: Modello *FE* completo: *wireframe*

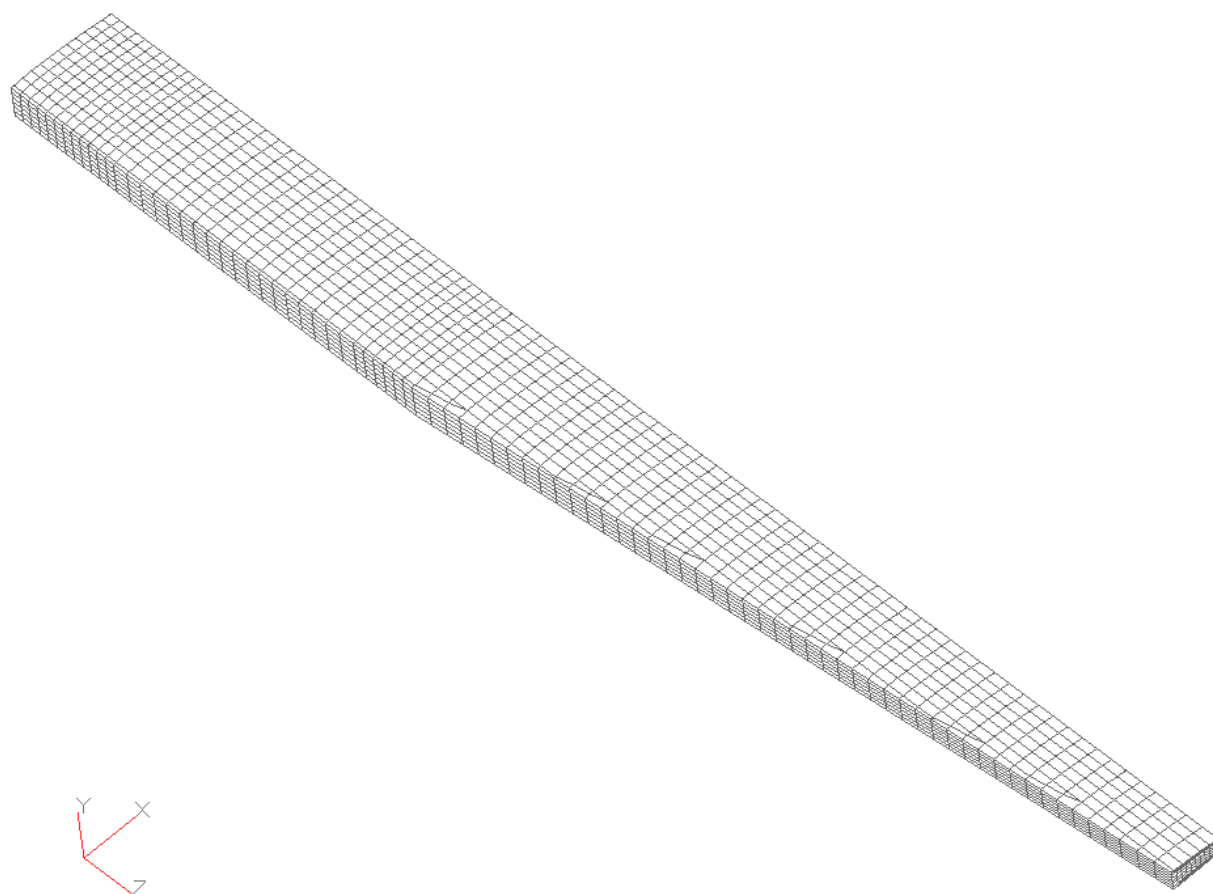


Figura 2.19: Modello *FE* completo: *hidden line*

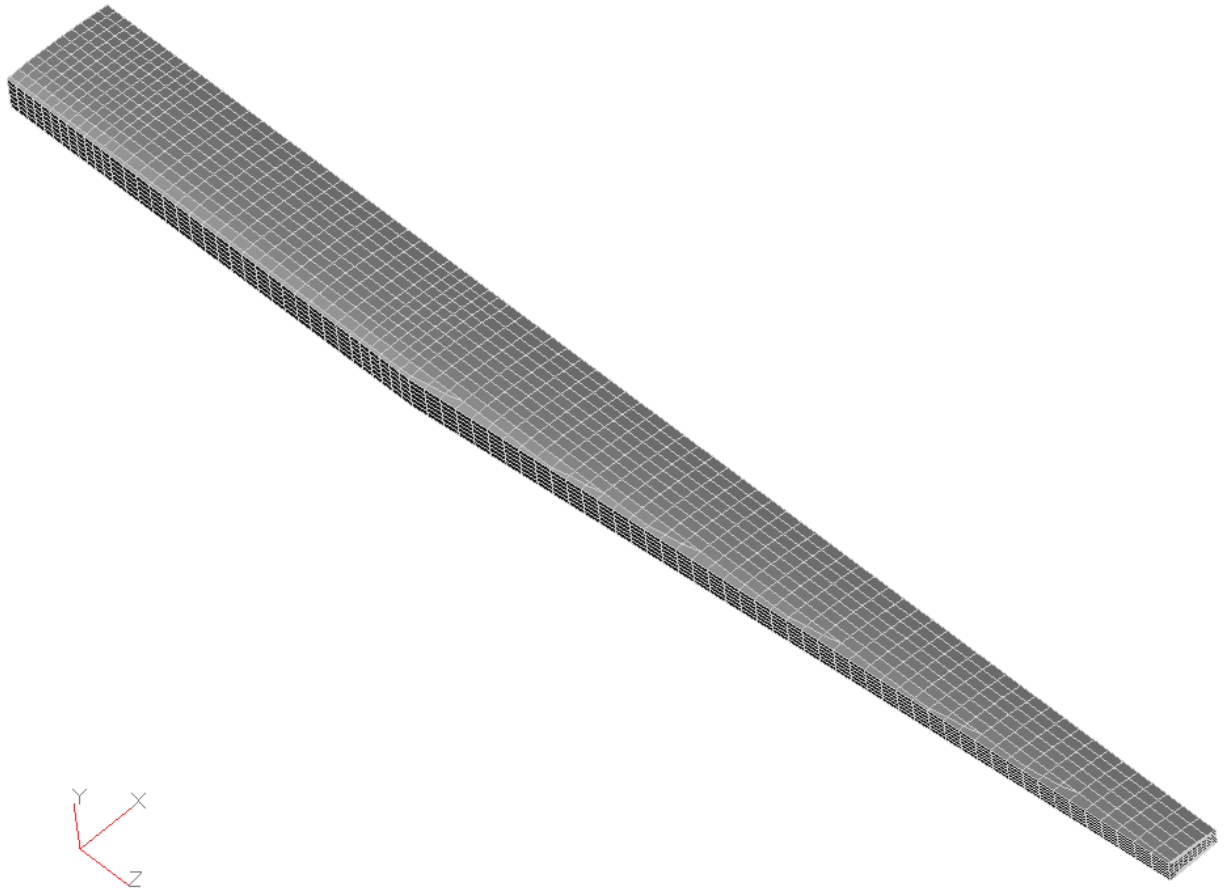


Figura 2.20: Modello *FE* completo: *shaded*

CAPITOLO 3

DEFINIZIONE DEI CARICHI E DEGLI AMMISSIBILI

3.1 Introduzione

Le condizioni di carico che la struttura deve dimostrare di poter sopportare nel corso delle prove di certificazione vengono definite nelle normative americane *FAR (Federal Aviation Regulations)*, riprese dalla comunità europea e riformulate nelle *CS (Certification Specifications)*, documento redatto dall'Agenzia Europea per la Sicurezza Aerea (*EASA*). La norma *CS-25*, relativa a velivoli commerciali da trasporto per più di nove passeggeri, distingue i carichi in:

- *limite*: massimi raggiungibili durante la vita operativa
- *estremi*: ottenuti moltiplicando i carichi limite per un fattore di sicurezza.

Qualora non sia detto esplicitamente il contrario, si adotterà un fattore di sicurezza $FdS = 1.5$.

L'aeromobile deve essere in grado di sopportare una condizione limite senza deformazioni permanenti che ne penalizzino le caratteristiche operative ed una estrema per almeno tre secondi senza collasso. Le condizioni di carico limite ed estremo sono caratterizzate come *carichi statici*, ovvero ipotizzati costanti nel tempo.

Nel caso invece di carichi variabili le normative si riferiscono a *sollecitazioni affaticanti*, di entità minore rispetto alle limite e ripetute per tutta la vita operativa dell'aeromobile. Per tali tipologie di sollecitazione deve essere dimostrata l'elevata improbabilità nel corso di tutta la vita operativa di un cedimento catastrofico a causa di carichi affaticanti, per tutte le parti principali del velivolo.

Viene valutata la resistenza statica residua, sulla scorta degli spettri di carico tipici della missione svolta e, quando presenti, dei dati storici relativi ad aerei di architettura simile, e verificato che questa stia, in ogni momento della vita del manufatto, al di sopra dei carichi definiti di *fail safe*, di norma uguali alle sollecitazioni limite.

3.2 Classificazione dei carichi

È possibile elencare i carichi agenti sul velivolo in diversi modi, ciascuno dei quali li distingue per un aspetto peculiare. In letteratura si trovano testi fondamentali ((18), (14), (21) sono alcuni) che aiutano nel compito di individuarne le diverse tipologie. Ad esempio si può porre l'attenzione sulla *natura* dei carichi in gioco, che possono essere:

1. **Aerodinamici:** *portanza e resistenza* agenti sulle superfici esterne del velivolo e *spinta* esercitata dall'impianto propulsivo
2. **Di massa:** agiscono su ogni massa della struttura per effetto della gravità e di altre accelerazioni alle quali il sistema può essere soggetto
3. **Al suolo:** *reazione verticale* ed *attrito* che gravano sulle ruote dei carrelli quando toccano terra

Sebbene importante per inquadrare il problema, questa classificazione rimane però puramente teorica. Una, più utile nella pratica, viene realizzata dalle normative (vedi (12)) e concerne il momento della missione nel quale avviene la condizione di carico:

1. In volo:
 - Simmetriche
 - Non simmetriche
2. Al suolo:
 - Atterraggio
 - Rullaggio
 - Corsa frenata
 - Operazioni a terra

Il panorama delineato da questo semplice elenco è in realtà vasto e articolato: ogni elemento della lista è rappresentato da una moltitudine di casi più o meno complessi che possono sollecitare parti diverse della struttura in modo differente, alcune delle quali possono

3.3 Carichi in volo

3.3.1 Condizioni simmetriche

Stazionarie o *balanced conditions*

Si definiscono così le manovre svolte in modo da non avere nessuna accelerazione angolare sull'aeromobile; due tipici esempi sono la virata (*wind-up turn*) e la richiamata (*pull-up*). Per determinare le forze agenti sul velivolo si usano le classiche equazioni del volo livellato, correggendo il peso del velivolo col fattore di carico:

$$L \cdot \cos \alpha + D \sin \alpha = n_z \cdot W \quad ; \quad D \cdot \cos \alpha - L \sin \alpha = n_x \cdot W + T_{eng} \quad .$$

Per determinare l'angolo di incidenza α occorre scrivere le equazioni di equilibrio dell'aereo in volo livellato nelle quali compaia anche la forza in coda. Risolvendo il sistema nelle incognite α e *BTL* (*balancing tail load*) è possibile stabilire l'entità delle forze aerodinamiche agenti sul velivolo e quindi sull'ala.

Generalmente la virata e la richiamata producono i massimi carichi di progetto per manovre simmetriche. Occorrerà dunque prestare molta attenzione a queste condizioni di volo, che si trovano nei limiti del diagramma di manovra (punti da **A** ad **I** in figura 3.1).

Manovre di equilibratore o *pitching conditions*

Si tratta di manovre simmetriche, perché l'aeromobile percorre una traiettoria giacente sul proprio piano di simmetria, e non stazionarie, perché sviluppano accelerazioni angolari. La normativa illustra i due tipi di comando contemplati:

- Gradino di equilibratore (*Maximum pitch displacement*): in condizione di volo livellato alla V_A (punto A_1) il pilota dà un brusco comando a cabrare. Si analizzano i carichi che si sviluppano entro il massimo n_z per la condizione in esame (punto A_2)
- Manovra scontrata (*Checked manoeuvre*): in condizione di volo livellato a qualsiasi velocità tra V_A e V_D il pilota dà un comando sinusoidale a cabrare o a picchiare, che sviluppa il massimo fattore di carico per manovra a cabrare e fattore di carico nullo per la manovra a picchiare. La legge temporale del comando è definita dettagliatamente.

Ragionando sulle manovre appena descritte è facile convincersi che non sono critiche per la struttura alare. Infatti inducono in ala delle forze inferiori a quelle che si avrebbero nelle condizioni simmetriche stazionarie.

Per la manovra di gradino questo risulta evidente osservando quanto segue: siccome la V_A è definita entro due valori limite

$$V_{S1} \cdot \sqrt{n_{z,max}} \leq V_A \leq V_C \quad , \quad n_{z,max} = 2.5$$

ma poiché la curva di stallo tende ad abbassarsi in corrispondenza di $V_{S1} \cdot \sqrt{n_{z,max}}$, accade che il fattore di carico massimo in condizione di volo alla velocità di manovra è inferiore ad $n_{z,max}$ (vedi punti A_2 ed A_3 in figura 3.2).

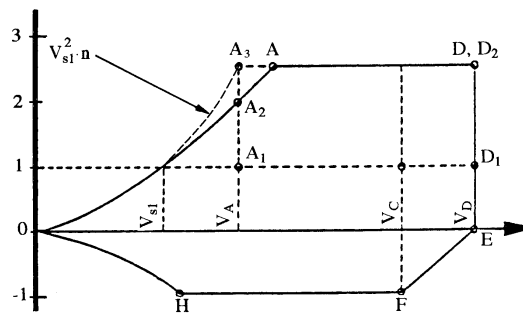


Figura 3.2: Effetto della comprimibilità sulla curva di stallo

Per quanto concerne invece la manovra scontrata, in essa si genera una forza ΔL_t che cambia segno durante l'evoluzione del moto:

$$L_{t,MS} = BTL + \Delta L_t \quad ,$$

dove il pedice MS indica la condizione di volo durante la manovra scontrata.

Nella fase in cui la variazione di portanza in coda è positiva (cioè verso l'alto) si avrebbe un incremento di fattore di carico poiché vale comunque la relazione

$$\frac{L_{wb,MS} + L_{t,MS}}{W} = n_{z,MS} \quad ,$$

ma poiché la normativa dice esplicitamente che deve essere $n_{z,MS} \leq n_{z,max}$ segue che:

$$L_{wb,MS} \leq L_{wb,stazionaria} \quad .$$

Raffiche discrete e continue

La normativa prevede che l'aeromobile possa essere investito da una raffica durante la missione. La richiesta è che la struttura resista alle forze interne generate da queste raffiche.

Si ipotizza che la forma della raffica sia del tipo *'uno meno coseno'* :

$$U(s) = \begin{cases} \frac{U_{ds}}{2} \cdot \left[1 - \cos\left(\frac{\pi \cdot s}{H}\right) \right], & \text{per } 0 \leq s \leq 2H \\ 0 & \text{per } s > 2H \end{cases}$$

con

$$U_{ds} = \frac{U_{ref} \cdot F_g \cdot H^{0.167}}{2.17} ,$$

relazione nella quale H è la dimensione della raffica, cioè la distanza tra valle e picco della funzione espressa in metri, e F_g un fattore di attenuazione funzione della quota operativa del velivolo. U_{ref} è la velocità della raffica di riferimento, dipendente della quota (vedi figura 3.3). Per determinare i carichi da raffica occorre svolgere un'analisi dinamica accurata per un numero sufficientemente grande di valori della dimensione della raffica H , tenendo in conto di tutte le caratteristiche aerodinamiche non stazionarie e di tutti i gradi di libertà significativi.

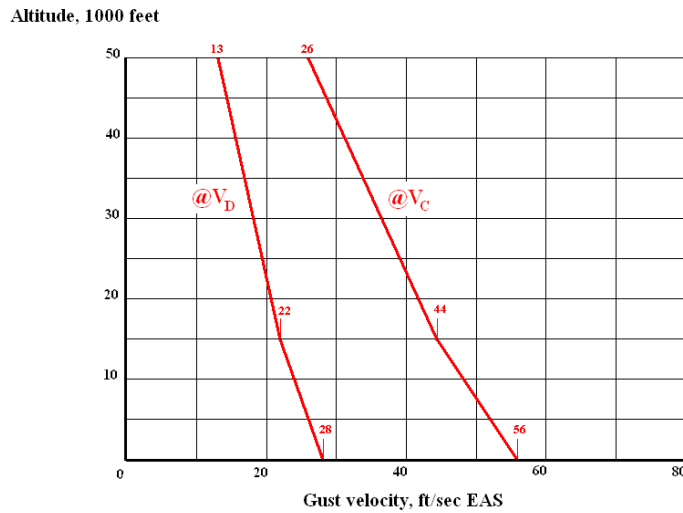


Figura 3.3: Velocità della raffica di riferimento

Per quanto riguarda le raffiche continue, le normative prevedono che debba essere analizzata la risposta dinamica dell'aeroplano ad un regime di turbolenza continua, sia verticale sia laterale. L'analisi dinamica richiesta deve tenere in conto le caratteristiche di aerodinamica non stazionaria e tutti i gradi di libertà strutturali significativi.

Dal punto di vista operativo, per una valutazione preliminare dell'entità dei carichi da raffica è utile servirsi della vecchia normativa la quale prevede di quantificare il fattore di carico da raffica prescindendo dagli effetti dinamici e dalla deformabilità dell'ala, di difficile valutazione in fase di avanprogetto.

A tale scopo si richiama la *formula di Pratt*:

$$n_{z\,gust} = 1 \pm \frac{K_g \cdot U_{de} \cdot V_{EAS} \cdot C_{N\alpha} \cdot S}{498 \cdot W} \quad , \quad (3.1)$$

nella quale compare la velocità equivalente dell'aereo V_{EAS} , la velocità della raffica U_{de} funzione della velocità equivalente e della quota, il carico alare, la pendenza della curva di forza verticale al variare dell'incidenza $C_{N\alpha}$ ed il fattore di attenuazione da raffica

$$K_g = \frac{0.88}{\frac{5.3}{\mu_g} + 1} \quad ,$$

funzione a sua volta del *mass ratio* o parametro di massa adimensionale

$$\mu_g = \frac{2 \cdot W}{\rho \cdot c \cdot S \cdot g \cdot C_{N\alpha}} \quad .$$

L'espressione del fattore deriva dalla soluzione dell'equazione del moto verticale dell'aereo soggetto ad una raffica *uno meno coseno* forzante, ad uno smorzamento aerodinamico e ad inerzia; se si usasse un profilo di raffica diverso, il fattore cambierebbe.

3.3.2 Condizioni non simmetriche

Rollio

Per le manovre di alettone devono essere verificate cinque condizioni con volo a fattore di carico nullo o pari a due terzi del massimo del diagramma di manovra:

- Condizione *stazionaria*: massima velocità angolare di rollio ed accelerazione nulla
- Massima accelerazione angolare (*Roll initiation*)

- Alla V_A una brusca deflessione degli alettoni fino a fondo corsa, tale da indurre una certa velocità angolare \hat{p}
- Alla V_C un comando tale da provocare la stessa \hat{p} del punto precedente
- Alla V_D la deflessione indurrà una velocità angolare non inferiore a $\frac{1}{3}\hat{p}$

Tutte le manovre di rollio possono essere critiche per la torsione del cassone alare in quanto la forza esercitata dall'alettone è fortemente eccentrica rispetto all'asse di riferimento. Durante il moto, le coppie inerziale e di smorzamento aerodinamico equilibrano la coppia del comando, come risulta evidente impiegando il modello del moto di rollio ad un grado di libertà:

$$L_{\dot{p}} \cdot \dot{p} + L_p \cdot p = L_{\delta_a} \cdot \delta_a \quad .$$

Conoscendo le caratteristiche aeromeccaniche del velivolo è possibile determinare la massima velocità angolare p e la massima accelerazione angolare \dot{p} :

$$p_{max} = \frac{L_{\delta_a} \cdot \delta_a}{L_p} \quad , \quad \dot{p}_{max} = \frac{L_{\delta_a} \cdot \delta_a}{L_{\dot{p}}} \quad .$$

La accelerazione angolare \dot{p} indurrà un fattore di carico verticale variabile in apertura, la velocità angolare p una forza centrifuga (quantificabile con un fattore di carico laterale n_y) importante per le verifiche agli attacchi delle gondole motore:

$$n_z = n_{z,sym} \pm y\dot{p}/g \quad , \quad n_y = yp^2/g \quad .$$

Per quanto riguarda una eventuale raffica che induca un moto di rollio nell'aeromobile, la condizione da analizzare è quella di una raffica discreta che induca il massimo fattore di carico su un'ala ed i quattro quinti di questo sull'altra.

Imbardata

La manovra richiesta dalla normativa è, per volo livellato a qualsiasi velocità tra V_{MC} (minima di controllo con un motore inoperativo) e V_D , con angolo di derapata β nullo:

1. Una brusca deflessione del timone fino a fondo corsa o, alternativamente, limitata dallo sforzo del pilota

2. Il comando esercitato al punto precedente viene tenuto fino al raggiungimento dell'equilibrio
3. Una volta raggiunto l'equilibrio, il timone viene bruscamente riportato alla posizione iniziale

Questo tipo di manovra induce una coppia flettente significativa in fusoliera ed una forza sulla deriva, pertanto dovrebbe essere presa in considerazione per il dimensionamento di queste parti, inoltre risulta critica per le ali che hanno grosse masse (per esempio un serbatoio) vicino all'estremità o dotate di *winglets*. Deve altresì essere posta l'attenzione sugli attacchi del motore, sollecitati dalle relative forze d'inerzia.

3.4 Carichi a terra

3.4.1 Atterraggio

Le normative prescrivono un gruppo di condizioni da verificare in fase di atterraggio: su due e tre punti (*tail down landing* e *level landing*), su un carrello solo (*one gear landing*), condizione di rimbalzo sulla pista (*rebound*) e la situazione di emergenza. Per ciascuna di esse occorre mettere in equilibrio le forze ed i momenti agenti nel sistema con le risultanti di inerzia. La normativa specifica il campo delle velocità e dei pesi da considerare e fornisce indicazioni sulle forze e i momenti che agiscono sul velivolo da applicare in mancanza di analisi dedicate di dettaglio. Questo tipo di manovre sono normalmente critiche per il dimensionamento dei carrelli principale ed anteriore e dei relativi attacchi.

3.4.2 Condizioni al suolo

Si tratta di condizioni di carico che possono insorgere durante il moto dell'aereo a terra: il rullaggio, che sarà più o meno severo in base al grado di asperità del terreno, la corsa frenata, la sterzata in pista, le situazioni nelle quali l'aeromobile viene trainato, sollevato e quant'altro.

3.5 Scelta delle condizioni di volo e calcolo dei carichi

Tra tutte le condizioni del panorama delineato nei paragrafi precedenti, sono state scelte cinque di esse: manovre ad alta e bassa incidenza, fattore di carico massimo e peso dell'aereo al decollo ed a serbatoi vuoti (*zero fuel*), e raffica verticale discreta in crociera, per la quale si è calcolato il fattore di carico grazie alla formula 3.1. Inoltre si è presa in considerazione una configurazione in volo livellato a velocità di crociera, fattore di carico unitario e peso medio (21780 Kg) per ricavare la tensione ad 1g, come riferimento per un controllo sulla resistenza a fatica.

Nella tabella 3.1 vengono elencati per esteso i dettagli relativi ad ogni condizione di volo scelta per il dimensionamento, mentre nella tabella 3.2 vi sono i valori numerici scelti per i pesi, le velocità caratteristiche e la quota di crociera.

Peso	Velocità	FdC	Quota	Descrizione
$W_{Take\ Off}$	V_A	$n_{z\ max}$	0	Manovra ad α alta e peso massimo
$W_{Take\ Off}$	V_D	$n_{z\ max}$	0	Manovra ad α bassa e peso minimo
$W_{Zero\ Fuel}$	V_A	$n_{z\ max}$	0	Manovra ad α alta e peso massimo
$W_{Zero\ Fuel}$	V_D	$n_{z\ max}$	0	Manovra ad α bassa e peso minimo
$W_{Zero\ Fuel}$	V_C	$n_{z\ raffica}$	0	Raffica in crociera
$W_{metà\ crociera}$	V_C	1	$h_{crociera}$	Volo livellato a metà crociera

Tabella 3.1: Condizioni di volo scelte

V_A	175 kts	$W_{Take\ Off}$	22800 Kg
V_C	275 kts	$W_{Zero\ Fuel}$	20800 Kg
V_D	340 kts	$W_{metà\ crociera}$	21780 Kg
$h_{crociera}$	17000 ft	$n_{z\ max}$	2.5

Tabella 3.2: Valori numerici

Si procede dunque col calcolo delle sollecitazioni indotte sull'ala. Riferendosi alla figura 2.16, riportata per comodità nella 3.4, occorre valutare centina per centina le tre forze e la coppia ivi rappresentate.

Le condizioni di volo scelte devono essere valutate come carichi limite ed estremi, in quest'ultimo caso moltiplicando per il fattore di sicurezza. Ma poiché il solutore strutturale svolge le simulazioni in ambito *lineare*, è probabile che per quanto concerne le condizioni estreme fornisca dei risultati eccessivamente conservativi. Infatti le strutture rimangono di

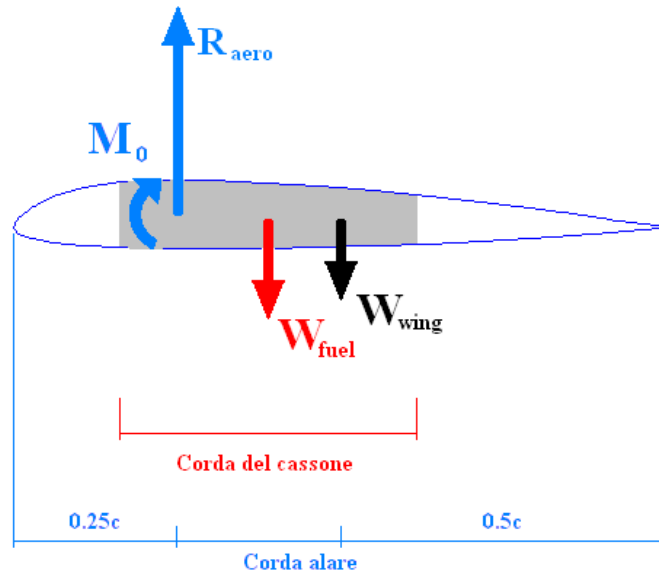


Figura 3.4: Forze agenti sul profilo

norma nella zona di proporzionalità tra carichi e tensioni fintanto che la sollecitazione sta al di sotto del valore limite, ma per le condizioni estreme il sistema si trova in uno stato di non proporzionalità (si veda la figura 3.5). Il modo con cui arriva dal punto **L** al punto **E** in figura non è noto a priori, si può soltanto affermare che $\sigma_{lim} \leq \sigma_{ult} \leq 1.5 \cdot \sigma_{lim}$ (punto **E'**).

Una soluzione lineare, di fatto, fornisce delle tensioni che differiscono dalle sollecitazioni per un fattore sempre uguale. Dunque, se $\hat{\sigma}$ è la tensione per una certa condizione limite, richiedere un'altra analisi con la corrispondente condizione estrema porterebbe soltanto a una soluzione pari a $FdS \cdot \hat{\sigma}$.

In base a quanto si è detto le soluzioni lineari forniranno un valore della tensione *superiore* a quello effettivo, e verificare che questo stia al di sotto di un ammissibile conduce ad un risultato conservativo. Per avere una stima migliore della risposta reale, occorrerebbe predisporre una simulazione *non lineare*; per il presente lavoro, comunque, si svolgeranno le analisi sul modello caricato al limite, dopodiché si ricaveranno le tensioni estreme moltiplicando le prime per il fattore di sicurezza. Si rimanda al paragrafo 6.1 per una discussione sulle simulazioni non lineari con *NASTRAN*.

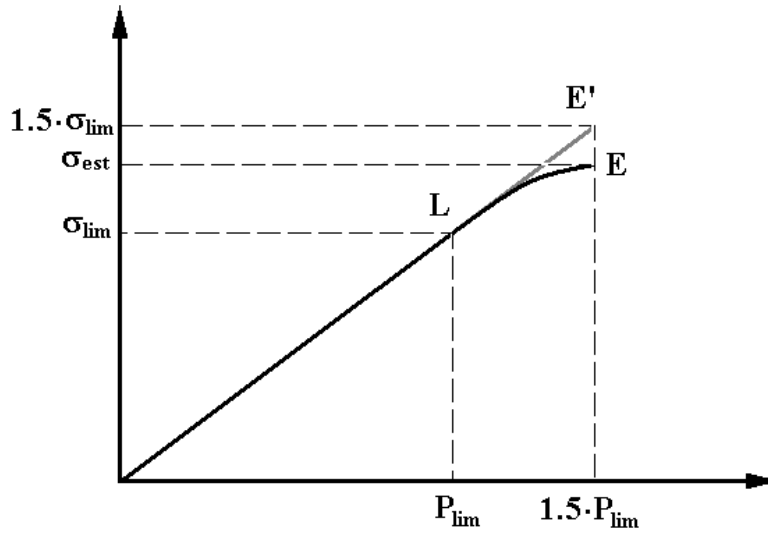


Figura 3.5: Andamento delle tensioni in funzione delle forze applicate

3.5.1 Calcolo delle risultanti aerodinamiche

Si è fatto uso del metodo approssimato sviluppato dal Diederich, il quale permette di stimare la distribuzione di portanza lungo l'apertura.

Si parte da una forma in pianta trapezoidale alla quale si riconduce l'ala effettiva, mantenendone inalterati i valori di apertura, corda al *tip* e superficie bagnata ed assumendo un'ala equivalente col bordo d'attacco dritto (si veda la figura 3.6). Il metodo, lungo l'apertura adimensionalizzata $\eta = 2y/b$, calcola i contributi $L_A(\eta)$ ed $L_B(\eta)$ alla portanza, chiamati rispettivamente distribuzione *aggiuntiva*, che equilibra il peso del velivolo, e *basica*, dovuta allo svergolamento alare ε , noto il coefficiente di portanza dell'aereo completo

$$C_L = 1.05 \cdot \frac{n_z \cdot W}{\frac{1}{2} \cdot \rho \cdot V^2} \quad ,$$

per calcolare il quale si è cautelativamente tenuto conto di un incremento del 5% della portanza necessaria a causa di una eventuale deportanza in coda. Una volta trovate le due distribuzioni, si ottiene, nota la corda media \bar{c} :

$$C_l(\eta) \cdot c(\eta) = C_L \cdot \bar{c} \cdot L_A(\eta) + \varepsilon \cdot C_{L\alpha} \cdot \bar{c} \cdot L_B(\eta) \quad .$$

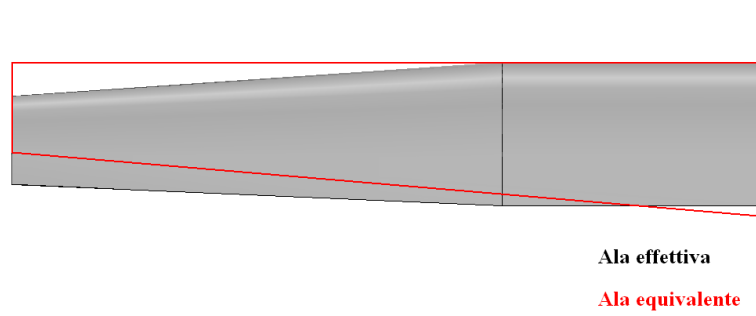


Figura 3.6: Forma in pianta equivalente per l'applicazione del metodo del Diederich

Il risultato ottenuto viene in seguito trasformato in forze che gravano su ogni centina, calcolate secondo la relazione:

$$R_{aero}(\hat{y}) = q \cdot \int_{\hat{y}-\frac{\Delta y}{2}}^{\hat{y}+\frac{\Delta y}{2}} C_l(\eta) \cdot c(\eta) \cdot d\eta = q \cdot C_l(\hat{y}) \cdot c(\hat{y}) \cdot \Delta y \quad ,$$

nella quale $q = \frac{1}{2}\rho V^2$ è la pressione dinamica, \hat{y} la coordinata del setto e Δy la distanza tra metà baia a monte e metà baia a valle (si veda la centina di coordinata \hat{y}_2 in figura 3.7). Dunque si prende il valore di portanza in corrispondenza della coordinata della centina e lo si suppone costante su tutta la porzione di superficie interessata.

Nelle centine che delimitano baie di lunghezza diversa (coordinata \hat{y}_1 in figura), la distribuzione aerodinamica ridotta a risultante determina anche una coppia di trasporto flettente in apertura; questo avviene ad esempio nella centina al *kink* o all'estremità, perchè a causa della diversa apertura tra cassone e superficie portante rimane una certa porzione di superficie alare i cui carichi vengono presi dall'ultima centina. Nel lavoro svolto si è tenuto conto soltanto di quest'ultima e supposto nulli tutti gli altri momenti aerodinamici di trasporto.

3.5.2 Calcolo del momento aerodinamico

Dovuto allo sbilanciamento delle pressioni in corda, l'entità del momento a portanza nulla M_0 si calcola con la seguente

$$M_0(\hat{y}) = q \cdot c^2(\hat{y}) \cdot \Delta y \cdot C_{m0} \quad ,$$

nella quale si è posto $C_{m0} = 0.013$, valore di riferimento ((5)) per i profili utilizzati.

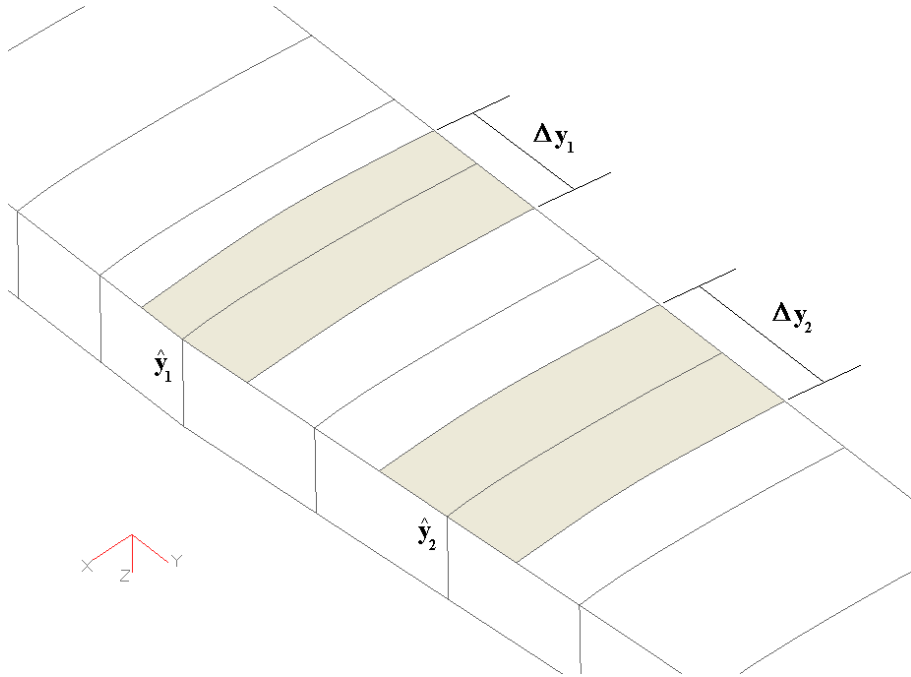


Figura 3.7: Superfici tra centine, semibaie diverse (\hat{y}_1) e uguali (\hat{y}_2)

3.5.3 Calcolo della risultante di massa strutturale e combustibile

Per la distribuzione di peso strutturale si suppone che vari lungo l'apertura proporzionalmente alla corda. Dunque :

$$p_{struct}(y) = \frac{n_z \cdot W_{tot\ ala}}{S} \cdot c(y) \quad ,$$

dove il peso totale dell'ala viene stimato con un modello sviluppato dalla NASA¹.

Si calcola invece il peso distribuito del combustibile con la relazione:

$$p_{fuel}(y) = \rho_{fuel} \cdot n_z \cdot c^2(y) \cdot ndsa(y) \quad ,$$

nella quale $\rho_{fuel} = 0.79 \cdot 10^3 \frac{kg}{m^3}$ è la densità del combustibile² ed $ndsa(y)$ è la distribuzione di superficie del cassone diviso la corda.

Le risultanti agenti su ciascuna centina si calcolano analogamente a come si è fatto per la risultante aerodinamica.

¹Beltramo, Michael N.; Trapp, Donald L.; Kimoto, Bruce W. e Marsh, Daniel P.: *Parametric study of transport aircraft systems cost and weight*. NASA CR-151970, anno 1977

²Long Range Aircraft, AEA Requirements, Dicembre 1989

3.5.4 Calcolo dei momenti di trasporto

Si è deciso di introdurre i carichi nel modello attraverso dei vincoli *MPC* aventi come polo un nodo a metà della corda della centina (si veda il paragrafo 2.3, in particolare la figura 2.14). Si deve dunque trasportare ogni risultante trovata su questo polo, operazione agevole note le distribuzioni di corda alare e strutturale dell'ala di riferimento (figura 3.8).

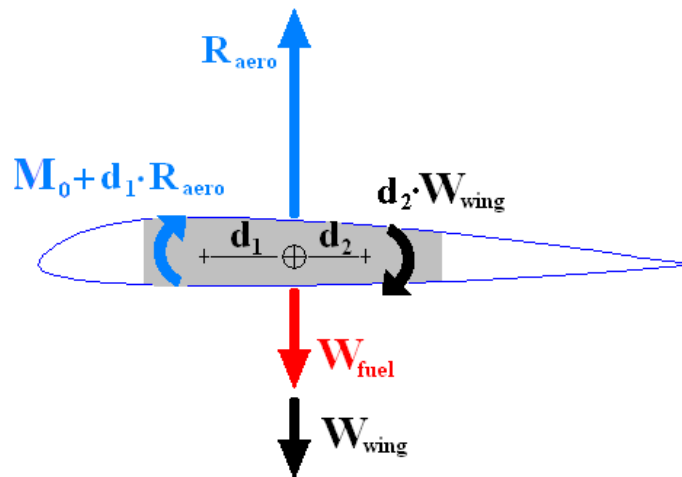


Figura 3.8: Calcolo dei momenti di trasporto

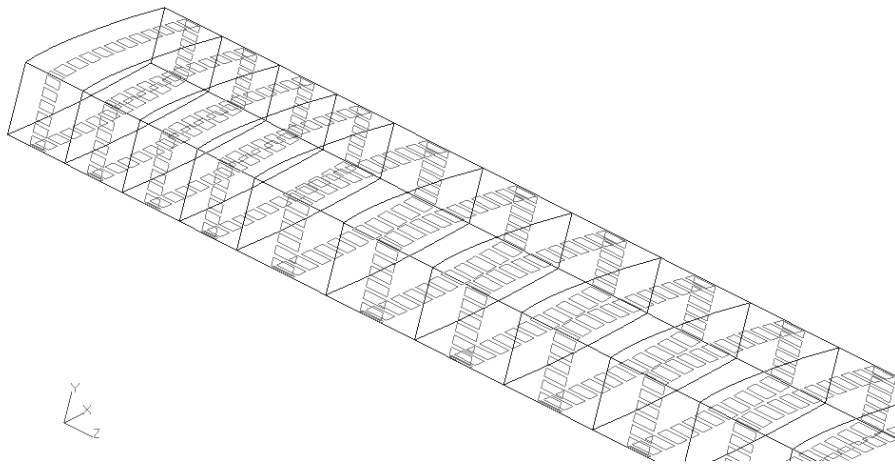


Figura 3.9: Gruppi di soluzione del ciclo

3.6 Scelta delle tensioni dimensionanti

Una volta caricato il modello e fatta partire la simulazione, il codice ad elementi finiti è in grado di fornire qualsiasi tensione in qualsiasi componente del modello. Ai fini dell'ottimizzazione della struttura, occorre indicare al *modeFrontier* quali tensioni controllare. La scelta deve essere rivolta alle sollecitazioni dominanti il problema, cioè quelle che hanno maggiori probabilità di portarsi al valore massimo consentito.

Pertanto si è deciso di controllare:

- gli sforzi di *trazione* nel rivestimento inferiore e di *compressione* nel superiore
- la tensione *taglio* nelle anime dei longheroni .

Lungi dal voler essere definitiva, questa scelta individua le sollecitazioni che in prima battuta potrebbero portare al collasso la struttura. Si tratta soltanto di una valutazione preliminare, al fine di scartare le configurazioni deboli e muoversi verso l'ottimo. In fasi più avanzate del progetto occorrerà tenere in conto molte altre tensioni e parti, così come bisognerà introdurre molte altre condizioni di volo.

Il *software* controlla i valori di tensione negli elementi in corrispondenza della mezzeria delle baie di controllo. Per svolgere questa operazione è stato necessario creare preliminarmente dei gruppi contenenti questi elementi, come si vede in figura 3.9. Per dettagli sulla definizione dei punti di controllo si rimanda all'appendice B.

3.7 Scelta degli ammissibili

3.7.1 Margini di sicurezza e controlli

Per ogni tensione investigata in ogni baia di controllo, il programma realizzato calcola il margine di sicurezza secondo la relazione

$$MS = 1 - \frac{\sigma_{applicata}}{\sigma_{ammissibile}},$$

nella quale si è posto σ , ma si intende caso per caso una tensione normale σ o tangenziale τ .

Sollecitazione	Ammissibile da verificare
<i>Trazione limite</i>	Snervamento
<i>Compressione limite</i>	Instabilità locale e globale, snervamento
<i>Trazione estrema</i>	Rottura
<i>Tensione di taglio estrema</i>	Instabilità a campo diagonale, rottura
<i>Compressione estrema</i>	Instabilità locale e globale, rottura

Tabella 3.3: Verifiche sui carichi

3.7.2 Caratteristiche del materiale

Si è scelta la lega di alluminio e rame **Al-2024-T3**, le cui caratteristiche principali sono riportate in tabella 3.4.

3.7.3 Ammissibili ad instabilità

Per ricavare le tensioni di instabilità a compressione locale e globale e la tensione tangenziale di instabilità a taglio sono state usate le seguenti relazioni:

$$\sigma_{cr,locale} = k \cdot \frac{\pi^2 \cdot E(\sigma)}{12 \cdot [1 - \nu^2(\sigma)]} \cdot \left(\frac{t}{b}\right)^2 \quad (3.2)$$

$$\sigma_{cr,euleriana} = c \cdot \frac{\pi^2 \cdot E_t(\sigma) \cdot \rho^2}{L^2} \quad (3.3)$$

$$\tau_{cr} = K_{ss} \cdot E \cdot \left(\frac{t}{h}\right)^2 \quad (3.4)$$

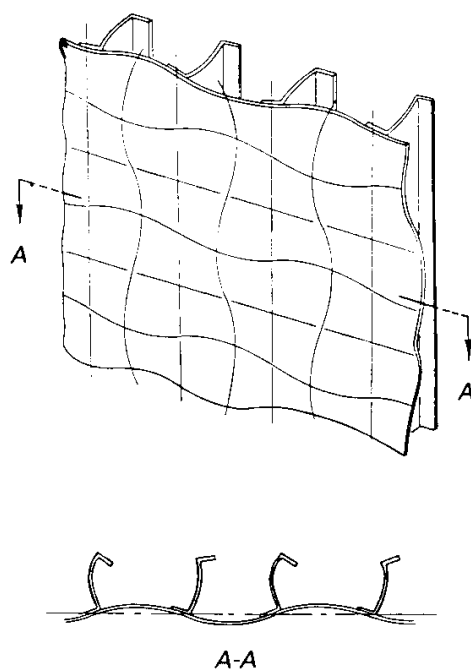


Figura 3.10: Instabilità iniziale del pannello irrigidito

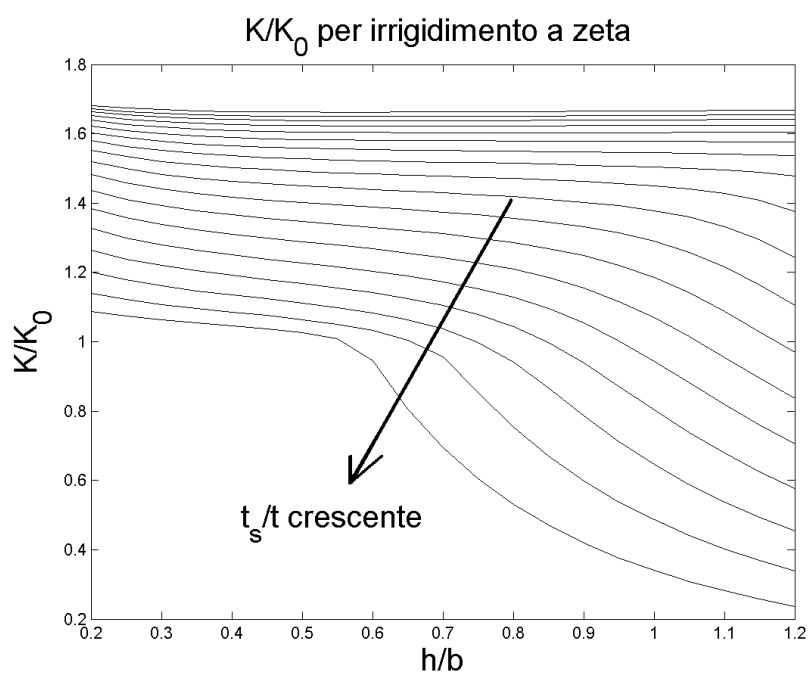


Figura 3.11: Diagramma K/K_0

E	72395 MPa
G	27600 MPa
$\nu_{elastico}$	0.33
$\nu_{plastico}$	0.5
ρ	$2.78 \frac{Kg}{dm^3}$
Ammissibili del materiale	
<i>Snervamento a trazione</i>	324 MPa
<i>Rottura a trazione</i>	441 MPa
<i>Snervamento a compressione</i>	269 MPa
<i>Rottura a compressione</i>	365 MPa
<i>Snervamento a taglio</i>	187 MPa
<i>Rottura a taglio</i>	275 MPa
Parametri di Ramberg Osgood	
σ_{07}	264 MPa
n	15

Tabella 3.4: Caratteristiche meccaniche della lega **Al-2024-T3**

L'equazione 3.2 fornisce la tensione critica locale del pannello irrigidito. Questo tipo di instabilità rappresenta un insieme di fenomeni - imbozzamento locale dei correnti ed instabilità della pelle - che avvengono in concomitanza, figura 3.10. Poiché è normalmente il tipo di instabilità che avviene per carichi più bassi, in letteratura viene anche chiamato *initial buckling*.

Nella formula compare il coefficiente k , funzione dei rapporti geometrici adimensionali della sezione, ed indice di come un elemento rettangolare di rivestimento delimitato da due correnti e due centine (*frame*) differisca in termini di carico critico da una piastra semplicemente appoggiata.

Riguardo alle caratteristiche del materiale, si è usata la notazione $E(\sigma)$, $\nu(\sigma)$ per mettere in luce le necessità di valutare il modulo di Young ed il coefficiente di Poisson in base all'entità della sollecitazione compressiva σ . Per comodità si usa riscrivere la relazione 3.2 in maniera che compaiano i valori di E e ν elastici ed accoppiare la dipendenza dalla tensione nel **fattore di correzione per la plasticità** η .

In letteratura esistono diverse espressioni empiriche per la valutazione del coefficiente η , per il presente lavoro si è scelta la relazione proposta da Gerard e Becker ((7))

$$\eta = \frac{1 - \nu_{el}^2}{1 - \nu^2(\sigma)} \cdot \left(\frac{E_s}{E} \right) \cdot \left[\frac{1}{2} + \frac{1}{4} \cdot \left(1 + 3 \cdot \frac{E_t}{E_s} \right)^{0.5} \right] ,$$

nella quale compare il coefficiente di Poisson effettivo

$$\nu(\sigma) = \nu_{pl} - (\nu_{pl} - \nu_{el}) \cdot \left(\frac{E_s}{E} \right)$$

ed i moduli secante $E_s(\sigma)$ e tangente $E_t(\sigma)$ del materiale, ciascuno funzione della tensione e calcolato con le formule di Ramberg Osgood

$$E_s(\sigma) = \frac{E}{1 + \frac{3}{7} \cdot n \cdot \left(\frac{\sigma}{\sigma_{07}} \right)^{n-1}}$$

$$E_t(\sigma) = \frac{E}{1 + \frac{3}{7} \cdot \left(\frac{\sigma}{\sigma_{07}} \right)^{n-1}}.$$

Sulla base delle osservazioni svolte, la 3.2 può essere riscritta in forma più compatta. Indicato con

$$K = \frac{k \cdot \pi^2 \cdot E}{12 \cdot (1 - \nu^2)}$$

un termine che raccoglie tutte le costanti della formula e con

$$K_0 = K_{(k=4)}$$

il valore di questo termine per una piastra semplicemente appoggiata sui quattro lati, è possibile riscrivere la relazione in questo modo:

$$\sigma_{cr,locale} = K \cdot \eta \cdot \left(\frac{t}{b} \right)^2, \quad (3.5)$$

e leggere il valore di $\frac{K}{K_0}$ direttamente nel diagramma, realizzato sulla base di simulazioni numeriche, riportato in figura 3.11.

La relazione 3.3 è il classico risultato della teoria delle strutture snelle caricate di punta. Secondo questo modo di instabilità, detto anche *globale* o *flessionale*, l'intero pannello irrigidito si comporta come se fosse una trave e si deforma come illustrato in figura 3.12; nella formula compare il coefficiente c dipendente da come sono vincolati i lati caricati del pannello,

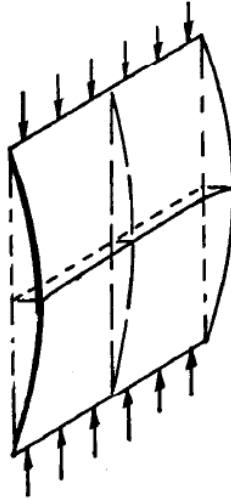


Figura 3.12: Instabilità globale del pannello irrigidito

il già menzionato modulo tangente, il raggio d'inerzia dell'unità ripetuta e la lunghezza L della baia.

Come coefficiente c si è usato il valore, intermedio tra appoggio ed incastro, di 1.5 suggerito in (21). Questa scelta è dovuta al fatto che le centine vincolano in qualche modo il pannello e questo ha una certa curvatura, mentre la 3.3 è valida per una piastra piatta.

La formula 3.4, tratta da (16), quantifica la tensione tangenziale critica di una piastra, di base d ed altezza h delimitata da flange e montanti, soggetta a taglio puro. Nella relazione, in modo del tutto analogo alla tensione critica iniziale, compare il coefficiente di vincolo K_{ss} , funzione del rapporto tra i lati (figura 3.13). È da notare che a denominatore deve essere messa caso per caso la dimensione del lato più corto.

3.7.4 Ammissibili a fatica

Per effettuare la verifica a fatica di un qualsiasi componente serve conoscere o ipotizzare la storia di carico alla quale esso verrà sottoposto durante la vita operativa. Una volta noto lo *spettro* delle tensioni si valuta il danneggiamento che queste provocano sulla struttura. In base alla filosofia di progetto scelta, si può tollerare la formazione di cricche da fatica a patto che non venga compromessa la sicurezza e che queste siano ragionevolmente limitate in numero: questo è l'approccio *fail safe* con la premessa, per esigenze economiche, della *durability*. Oppure si esclude a priori la nucleazione delle cricche abbassando le tensioni

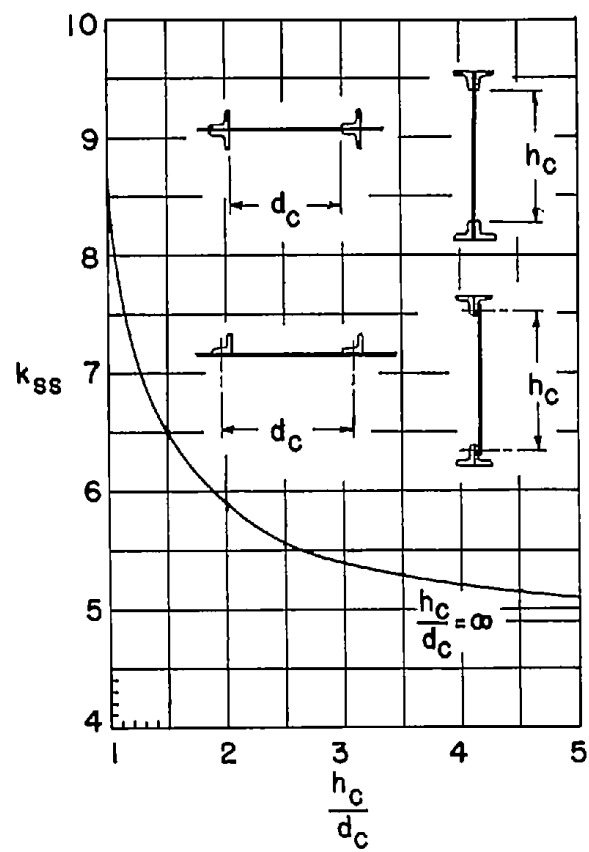


Figura 3.13: Coefficiente di vincolo della piastra caricata a taglio

alle quali il componente lavora, perdendo in termini di peso ma guadagnando in termini di semplicità d'ispezione; questo approccio è chiamato *safe life*.

Ad ogni modo, per il presente lavoro si tralasceranno questi metodi di valutazione, non avendo a disposizione un ragionevole modello degli spettri di carico. Poichè si riconosce l'importanza e l'impatto che ha il dimensionamento a fatica sul progetto di un cassone alare, si è voluto introdurre comunque una verifica, seppure statica.

Data la tensione alla quale la struttura portante lavora in condizioni di volo livellato e fattore di carico unitario, si verifica che essa non superi un valore di riferimento $\sigma_{amm\ fatica}$ in una baia vicino alla radice ed una vicino al *kink* nella zona ventrale:

$$\sigma_{1g} \leq \sigma_{amm\ fatica} = 85MPa \quad .$$

4.1 Introduzione al *modeFrontier*

Il *modeFrontier* è un ambiente di ottimizzazione multiobiettivo, pensato per integrare tra loro differenti strumenti di progetto assistito dal calcolatore (*Computer Aided Engineering*), ad esempio codici di analisi strutturale o fluidodinamica, permettendo di eseguire un processo impostato dall'utente nel quale vengono di volta in volta modificati i valori delle variabili di ingresso sulla scorta del controllo delle variabili in uscita, che possono rappresentare obiettivi dell'ottimizzazione oppure vincoli sulla bontà della soluzione. L'operatore definisce lo spazio nel quale agirà il codice (ad esempio un sistema fisico con le proprie leggi), e questo massimizzerà o minimizzerà le variabili in gioco, fornite coerentemente con lo spazio e le leggi specificate.

L'interfaccia utente del *modeFrontier* è un ambiente grafico di modellazione a blocchi, formato da nodi di esecuzione di applicazioni e di acquisizione (lettura di valori numerici da un *file* testuale) e trasferimento di dati. Il programma gestisce i flussi tra i blocchi applicative, intervenendo sui *file* di ingresso di ognuno. È possibile, con certi *software*, ad esempio **Matlab**, un collegamento dedicato ma qualsiasi applicazione si può comunque richiamare attraverso semplici programmi di poche righe di comando in *DOS*.

L'utilità di *modeFrontier* risiede nel fatto che, oltre a permettere di usare insieme campi di applicazioni differenti, fornisce l'utente di metodi di ottimizzazione e di opportuni strumenti di elaborazione che consentono di visualizzare l'evoluzione degli obiettivi nel corso del processo e di farne l'analisi statistica. Tra i criteri supportati spiccano gli algoritmi genetici, dotati del grosso vantaggio di poter effettuare calcoli di ottimo con più di un obiettivo. La generazione di algoritmi precedente, invece, nella quale si annovera il classico metodo del

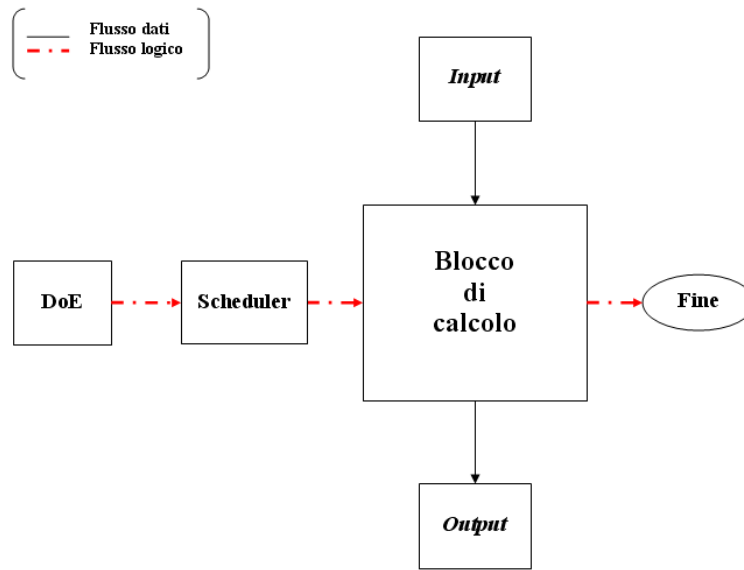


Figura 4.1: Schema semplificato di un progetto *modeFrontier*

gradiente coniugato, consente al più il conseguimento di uno solo; il progettista che si trovi nella necessità di massimizzare due o più funzioni obiettivo, qualora non disponesse degli algoritmi di nuova concezione, si dovrebbe accontentare di un'unica funzione che rappresenti una media pesata degli obiettivi desiderati.

Nell'ambiente del codice, è possibile distinguere il **flusso logico**, cioè la sequenza delle operazioni da realizzare, ed il **flusso di dati**, costituito da variabili interne e *file* esterni in ingresso e in uscita dai nodi. In figura 4.1 è riportato un diagramma semplificato di uno schema a blocchi di *modeFrontier*.

4.1.1 Flusso logico

Inizia col blocco di *Design of Experiments* (*DoE*), nel quale vengono definiti gli insiemi di variabili del sistema fisico che verranno usati per le prime simulazioni. Questi possono essere introdotti direttamente dall'utente oppure generati in maniera casuale o secondo uno schema fattoriale, che distribuisce i punti di partenza nello spazio delle soluzioni in modo più bilanciato di quanto non facciano i sistemi *random*. La scelta del *DoE* è funzionale di quello che si vuol fare: i fattoriali si usano nel caso gli esperimenti richiesti debbano essere usati per analisi statistiche o per addestrare una superficie di risposta in quanto coprono meglio

lo spazio senza addensare troppi punti in una sola zona (fenomeno del *clustering*, che inficia le valutazioni di questo tipo), mentre i randomici sono una buona base di partenza per un'ottimizzazione genetica in quanto hanno una maggiore varietà di valori ed arricchiscono il patrimonio di caratteristiche tra le quali l'algoritmo può scegliere.

Dopo il primo blocco vi è lo *scheduler*, nel quale si seleziona il metodo di analisi, a partire dalla semplice simulazione e calcolo del *DoE* fornito fino agli algoritmi multiobiettivo passando per i monobiettivo. Ciascun metodo ha i suoi parametri da impostare mediante questo nodo.

Segue il blocco di calcolo vero e proprio, nel quale viene richiamata una sequenza di operazioni scelte dall'operatore; non si tratta necessariamente di un flusso unico: lo schema potrebbe complicarsi con blocchi condizionali e sincronizzazione di flussi paralleli.

Il flusso logico termina con un blocco di fine delle operazioni.

4.1.2 Flusso dati

I dati possono essere variabili presenti nello spazio di lavoro di *modeFrontier* oppure qualsiasi tipo di *file* che serva per il funzionamento delle applicazioni o per l'estrazione di variabili numeriche.

Nell'ambiente del programma è possibile introdurre delle variabili, di ingresso e di uscita, scalari oppure vettoriali. I *file* in gioco possono essere trasferiti da un nodo all'altro, ed è altresì possibile estrarre oppure introdurre valori numerici di interesse da un *file* di testo.

4.2 Schema logico del ciclo

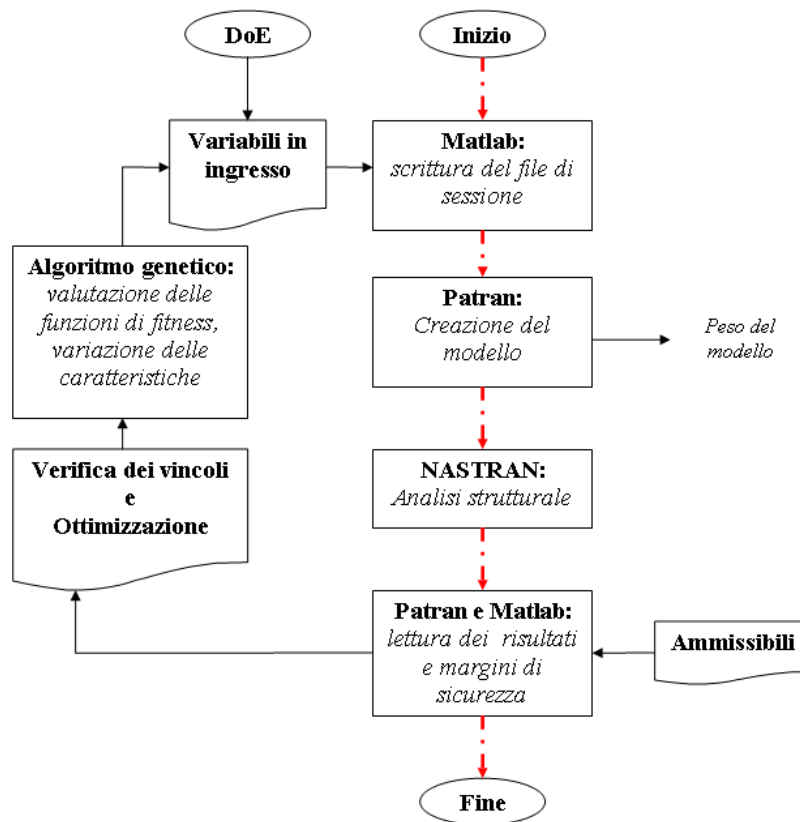


Figura 4.2: Diagramma di flusso dell'ottimizzatore

Per svolgere l'ottimizzazione, *modeFrontier* realizza automaticamente le azioni che svolgerebbe un qualsiasi utente: parte da dati di *input*, realizza il modello tridimensionale della struttura, esegue **NASTRAN**, analizza i risultati ed in base ad essi modifica il modello e riprende da capo. Si riporta in figura 4.2 uno schema del processo svolto dall'ottimizzatore.

La creazione della geometria viene realizzata da **MATLAB** che scrive il *file* di sessione di **Patran** sulla base dei dati forniti in ingresso. Verrà quindi generato un modello avente in ogni caso la stessa forma in pianta, nel quale però potrebbero esserci più o meno centine e correnti in funzione delle spaziature fornite in *input*. Una volta creato il *file* di sessione, viene fornito al nodo che richiama **Patran** la prima volta (fase pre-processo), il quale provvede in seguito alla realizzazione degli elementi finiti per l'analisi vera e propria. Definite le dimensioni geometriche ed il materiale costituente, è possibile calcolare il peso complessivo ed introdurlo sottoforma di variabile nell'ambiente di *modeFrontier*.

Dopo questa prima fase si esegue la simulazione con **NASTRAN**, le cui uscite vengono fornite al secondo nodo **Patran** (post-processo) nel quale si leggono le tensioni di interesse e fornite ad un altro nodo: un codice **Matlab** raccoglie i dati e, calcolati gli ammissibili per ogni tipo di sollecitazione, li riorganizza determinando i margini di sicurezza, alcuni dei quali verranno minimizzati (si veda più avanti), fermo restando che tutti devono essere positivi.

Terminato il processo descritto, l'ottimizzatore provvede automaticamente a modificare i dati geometrici in ingresso in base ai meccanismi propri degli algoritmi genetici: delle configurazioni appartenenti alla generazione precedente viene valutata la 'forma fisica' (funzione di *fitness*, in realtà è un indice matematico della bontà della soluzione in base al rispetto dei vincoli ed al valore delle funzioni obiettivo in relazione ai valori delle stesse per tutta la popolazione) e le più meritevoli, dette soluzioni *dominanti*, hanno una maggior probabilità di partecipare alla generazione successiva, che sarà il risultato di mutazioni o incroci delle famiglie precedenti. Sebbene il meccanismo di variazione dei parametri sia casuale, il criterio di selezione non lo è perchè si basa sulla funzione di *fitness* e quindi garantisce che il processo prosegua verso configurazioni migliori delle precedenti.

Una volta definito il nuovo insieme di variabili che caratterizza la configurazione, si riparte da un nuovo modello, e così via per un certo numero di generazioni o finchè il processo non sia giunto a convergenza.

Tutto il meccanismo descritto richiede per ogni configurazione pochi minuti di calcolo, al termine si ha un modello del quale è possibile dire quanto soddisfatti i requisiti. Questo strumento si profila pertanto particolarmente utile nelle prime fasi del progetto, nelle quali si cerca di valutare un gran numero di configurazioni nel minor tempo possibile.

Tutte le variabili interessate dal processo sono riportate in tabella 4.1. Alcune di esse vengono quasi sempre tenute costanti (ad esempio i fattori di proporzionamento di elementi oppure i fattori di scala delle aree dei *caps*), altre vengono fatte variare a seconda delle esigenze dell'utente.

Lo spessore delle centine è stato assunto costante per tutta l'apertura.

Gli spessori dei pannelli e delle anime e le caratteristiche degli irrigidimenti - spessore e altezza - variano linearmente lungo l'apertura. In un primo momento si è pensato di variare liberamente i valori alla radice, al *kink* ed al *tip* di ogni caratteristica geometrica. Questa soluzione è stata abbandonata quando è risultato evidente che, senza nessun vincolo sulla rastremazione, era frequente ottenere soluzioni nelle quali una dimensione x al *kink* fosse stata maggiore della corrispondente alla radice. La via di mettere un vincolo su questi valori si è rivelata da subito impraticabile: per la quantità di dimensioni in gioco la probabilità che almeno una non rispetti il vincolo $x_{root} \geq x_{kink}$ è inaccettabilmente elevata.

Per ovviare al problema, vengono forniti in ingresso non più i tre valori alle tre stazioni principali, bensì un valore minimo costante, il valore alla radice e due coefficienti, $0 \leq \alpha \leq 1$

Variabili di <i>input</i> del ciclo	Descrizione
$b_{irr\ sup}, b_{irr\ inf}, L_{in}, L_{out}$	Spaziature di correnti e centine
$t_{centine}$	Spessore delle centine
$t_{min\ sup}, t_{root\ sup}, \alpha_{pan\ sup}, \beta_{pan\ sup}$	Spessori dello <i>skin</i> superiore
$t_{min\ inf}, t_{root\ inf}, \alpha_{pan\ inf}, \beta_{pan\ inf}$	Spessori dello <i>skin</i> inferiore
$t_{min\ ant}, t_{root\ ant}, \alpha_{anima\ ant}, \beta_{anima\ ant}$	Spessori dell'anima del longherone anteriore
$t_{min\ post}, t_{root\ post}, \alpha_{anima\ post}, \beta_{anima\ post}$	Spessori dell'anima del longherone posteriore
$h_{min\ sup}, h_{root\ sup}, \alpha_{str\ sup}, \beta_{str\ sup}$	Altezze dei correnti superiori
$h_{min\ inf}, h_{root\ inf}, \alpha_{str\ inf}, \beta_{str\ inf}$	Altezze dei correnti inferiori
$t_{str\ min\ sup}, t_{str\ root\ sup}, \alpha_{t\ str\ sup}, \beta_{t\ str\ sup}$	Spessori degli irrigidimenti dorsali
$t_{str\ min\ inf}, t_{str\ root\ inf}, \alpha_{t\ str\ inf}, \beta_{t\ str\ inf}$	Spessori degli irrigidimenti ventrali
$scala_{ant\ sup}, scala_{ant\ inf}$ $scala_{post\ sup}, scala_{post\ inf},$	Coefficienti di scala delle aree delle flange dei longheroni
$D_{x\ cap, i}, D_{y\ cap, i}$	<i>Offset</i> delle flange dei longheroni
$\frac{d}{h\ sup}, \frac{d}{h\ inf}, \frac{d}{h\ ant}, \frac{d}{h\ post}$	Fattori di proporzionamento di irrigidimenti e flange
$rib_{AR}, spar_{AR}, rib_{PARAM}$	Fattori di proporzionamento degli elementi finiti
$check_{IN}, check_{OUT}$	Numero di baie <i>inboard</i> ed <i>outboard</i> da controllare

Tabella 4.1: Variabili in ingresso nel ciclo di ottimizzazione

e $0 \leq \beta \leq 1$, che permettono di ricavare i valori al *kink* ed all'estremità mediante le seguenti relazioni:

$$x_{kink} = x_{root} \cdot \left[\alpha + \frac{x_{min}}{x_{root}} \cdot (1 - \alpha) \right] \quad (4.1)$$

$$x_{tip} = x_{kink} \cdot \left[\beta + \frac{x_{min}}{x_{kink}} \cdot (1 - \beta) \right] \quad (4.2)$$

In questo modo, dato il valore alla radice della variabile, al limite si può avere che rimanga costante in tutta l'apertura, caso in cui è $\alpha = 1, \beta = 1$, oppure che dal *kink* in poi sui mantenga fino al *tip* sul valore minimo, caso in cui è $\alpha = 0, \forall \beta$; in ogni caso i valori al *kink*, per come sono stati definiti, non saranno in nessun caso maggiori dei corrispondenti alla radice. Si veda al riguardo la figura 4.3.

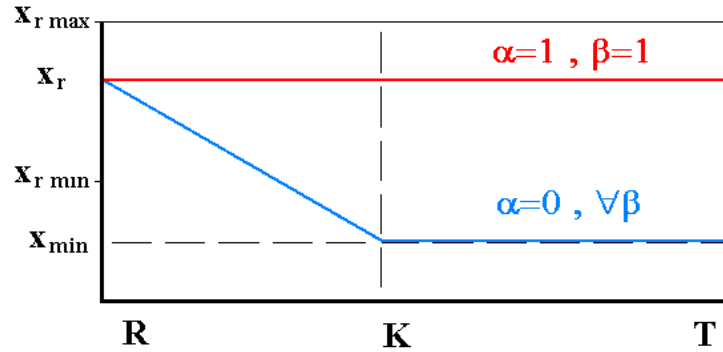


Figura 4.3: Diagramma illustrativo del metodo di inserimento delle dimensioni rastremate

Al fine di limitare il numero di variabili in gioco, per alcune caratteristiche geometriche si è preferito introdurre un adimensionale costante. Le aree delle flange dei longheroni, ad esempio, sono calcolate in base alle aree dei correnti, secondo la relazione:

$$A_{ant/post,sup/inf} = scala_{ant/post,sup/inf} \cdot A_{irr\ sup/inf} \quad .$$

Allo stesso modo sono state definite le basi delle sezioni dei correnti dorsali e ventrali e delle flange, con i coefficienti $\frac{d}{h_{sup}}$, $\frac{d}{h_{inf}}$, $\frac{d}{h_{ant}}$, $\frac{d}{h_{post}}$.

Non essendo possibile controllare ogni baia del modello, il cui numero totale varia al variare delle spaziature delle centine, si stabilisce un certo numero di stazioni *inboard* ed *outboard* da controllare (parametri $check_{IN}$ e $check_{OUT}$) ed il codice calcola a quali baie del modello corrispondano. Se ad esempio si avessero quattro stazioni, queste sarebbero al 25%, 50%, 75% e 100% in ogni caso, sia per i modelli con dieci baie sia per quelli con venti, cambierebbe però il numero delle baie da analizzare. Il programma **Matlab** svolge questa operazione e determina quelle da verificare volta per volta. Ad ogni modo, questi parametri sono stati tenuti costanti per tutte le simulazioni, in quanto cambiarli potrebbe comportare di dover cambiare la struttura dei nodi di acquisizione dei dati in uscita da **NASTRAN**.

Lo schema completo del ciclo, così come appare attraverso l'interfaccia grafica del *mode-*

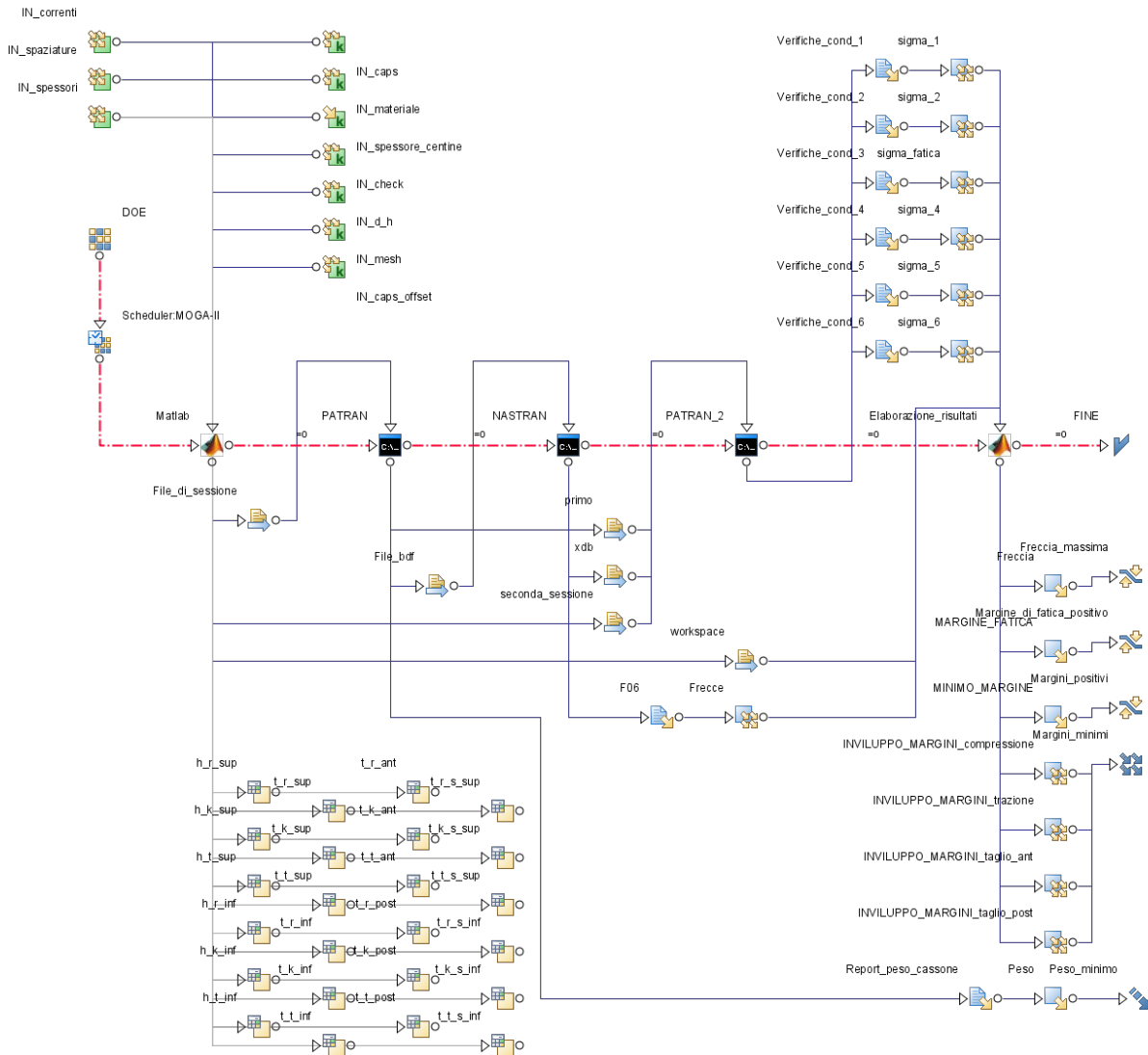


Figura 4.4: Schema completo del progetto *modeFrontier*

Frontier, è riportato in figura 4.4. In questa è possibile vedere il flusso logico, rappresentato da una tratteggiata rossa, attraverso la quale si passa dal *DoE* allo *scheduler*, al primo ed al secondo nodo **Matlab** passando per i nodi **Patran** e **NASTRAN**, al nodo di fine del processo.

In alto a sinistra ci sono i blocchi (in verde) di immissione dei dati, in basso a sinistra i blocchi (in giallo) di variabili ausiliarie, cioè non coinvolte direttamente nel ciclo – sono collegate solo in ingresso. Per comodità dell'utente, infatti, vengono calcolate le dimensioni al *kink* ed al *tip* con l'ausilio delle equazioni 4.1 e 4.2, in modo da avere un resoconto più diretto delle soluzioni trovate, in quanto è più immediato rendersi conto del valore di una dimensione piuttosto che di un coefficiente adimensionale quali sono α e β .

In alto a destra è possibile vedere i blocchi (celeste) di estrazione dei risultati, nei quali vengono lette le tensioni da una serie di *file* di testo prodotti da **Patran**, detti *report*, contenuti ciascuno i valori degli sforzi nelle baie di controllo relativi alle sollecitazioni dimensionanti per una condizione di carico. È possibile fornire al *modeFrontier* un esempio di come saranno fatti i *report file* ed in quali posizioni, in termini di righe e colonne, andare a cercare i risultati voluti. La prima colonna di nodi di colore celeste compie questa operazione, mentre la seconda trasforma i valori appena estratti in vettori di variabili.

In basso a destra, infine, vi è il blocco di verifica dei vincoli e di ottimizzazione. Le variabili scelte per questo blocco vengono ottimizzate, ad esempio il peso così come alcuni margini di sicurezza vengono minimizzati, oppure viene verificato che non superino un certo valore (vincoli o *constraints*), come la freccia massima al *tip*, che non deve stare al di sopra di un certo valore, oppure i margini di sicurezza che devono essere tutti positivi.

In appendice A vengono forniti ulteriori dettagli sul ciclo di ottimizzazione e sulle operazioni svolte dalle singole applicazioni.

4.3 Scelte di vincoli e funzioni obiettivo

Per permettere al processo di evolvere verso soluzioni migliori di quelle date in partenza occorre definire un opportuno insieme di vincoli e obiettivi. La scelta non è univoca, in essa entrano in gioco punti di vista spesso soggettivi.

Indiscutibilmente lo scopo più importante da perseguire è la minimizzazione del peso complessivo della struttura, senz'altro contemplato nel processo.

Un altro obiettivo è ottenere un cassone che abbia una certa rigidezza flessionale, quantificabile con la deflessione massima all'estremità. Questa condizione viene posta sottoforma di vincolo sulla freccia, che non deve essere superiore ad un certo valore, pena la non accettabilità della soluzione.

Per qualsiasi condizione di carico alla quale venga sottoposta la struttura, questa non deve sviluppare tensioni al di sopra degli ammissibili. Ciò viene imposto nella forma di un vincolo sulla positività di tutti, oppure, il che è lo stesso, del minore tra tutti i margini di sicurezza.

L'ultima condizione da imporre deve introdurre in qualche maniera la necessità di sfruttare al meglio il materiale. Infatti non è sufficiente che il peso sia il minimo possibile, occorre anche che i margini di sicurezza, che comunque non devono essere negativi, siano abbastanza vicini al limite. Per spiegare meglio questo concetto, si supponga di avere due cassoni di egual peso con involucri di margini di sicurezza in apertura riportati in figura 4.5(a) e 4.5(b). Il primo cassone, sebbene abbia lo stesso peso, sfrutta peggio il materiale rispetto al secondo, che è da preferirsi per la maggior efficienza strutturale.

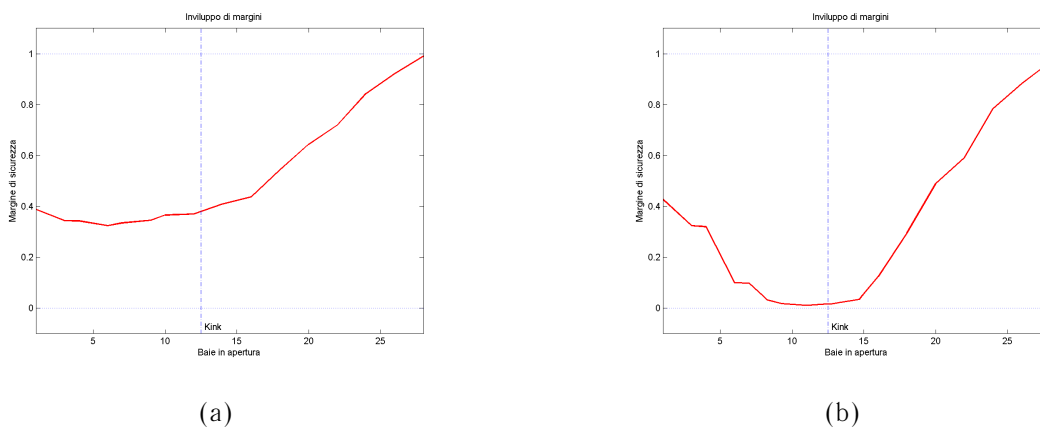


Figura 4.5: Esempio di valutazione dei margini di sicurezza

L'ultima condizione scelta per l'ottimizzazione è quindi la minimizzazione dei margini di sicurezza in apertura. Per evitare un eccessivo numero di funzioni obiettivo, deleterio per l'efficacia del processo che raggiungerebbe l'ottimo molto più lentamente, si impone la

condizione di minimo in tre punti lungo l'apertura, uno vicino alla radice e due, *inboard* ed *outboard*, vicini al *kink*, per ciascuna delle quattro tensioni dimensionanti. Le scelte illustrate sono riassunte nella tabella 4.2

Variabile	Vincolo o obiettivo
Peso	<i>Minimizzare</i>
Freccia masima	$\leq 10\%$
MS minore	≥ 0
MS in apertura	<i>Minimizzare</i>

Tabella 4.2: Vincoli e obiettivi del processo

CAPITOLO 5

ESEMPI DI PROGETTO OTTIMIZZATO DI STRUTTURE

Nel presente capitolo verranno illustrati alcuni risultati ottenuti con lo strumento realizzato. Si sono cercati dei casi esemplificativi di progetto strutturale ottimizzato. Per saggiare le potenzialità della procedura, un processo di ottimizzazione del passo dei correnti del cassone è riportato nel paragrafo 5.1; una seconda ed una terza simulazione, aventi l'intento di valutare rispettivamente la spaziatura e lo spessore ottimo delle centine, vengono illustrate nel paragrafo 5.3.

In tutti gli esempi che seguiranno sono stati assunti i parametri riportati in tabella 5.1 ed è stato usato l'algoritmo genetico *Multi Objective Genetic Algorithm* o *MOGAI*, il quale è pensato per una rapida convergenza alla frontiera di Pareto e permette di modificare l'influenza di alcuni parametri dell'algoritmo tra i quali mutazione e *crossover*.

5.1 Ottimizzazione del passo dei correnti

Come primo esempio si è impostato il calcolo di ottimizzazione del cassone alare avendo fissato tutti i parametri geometrici, tranne il passo dei correnti dorsali. Il passo dei correnti ventrali è stato assunto pari al 130% del passo dei dorsali. In tabella 5.2 sono riportati i dati in ingresso non specificati nella tabella 5.1.

L'incremento scelto per il passo è tale da dividere l'intervallo in 25 parti. Nei diagrammi tratti dall'ambiente di *modeFrontier*, si indica con puntini pieni le configurazioni che rispettano i vincoli imposti, viceversa con puntini vuoti le configurazioni *unfeasible*.

Dall'analisi dei diagrammi nelle figure da 5.1 a 5.5, realizzati con l'interfaccia grafica del *modeFrontier*, emergono diverse caratteristiche interessanti.

Variabile	Valore	Variabile	Valore
$t_{root\ sup}$	$4 \div 8\ mm$	$t_{root\ inf}$	$4 \div 8\ mm$
$t_{root\ ant}$	$4 \div 8\ mm$	$t_{root\ post}$	$4 \div 8\ mm$
$t_{str\ root\ sup}$	$4 \div 8\ mm$	$t_{str\ root\ inf}$	$4 \div 8\ mm$
$h_{root\ sup}$	$60 \div 80\ mm$	$h_{root\ inf}$	$60 \div 80\ mm$
α_i	$0 \div 1\ \forall i$	β_i	$0 \div 1\ \forall i$
$t_{min\ sup}$	1 mm	$t_{min\ inf}$	1 mm
$t_{min\ ant}$	1 mm	$t_{min\ post}$	1 mm
$t_{str\ min\ sup}$	1 mm	$t_{str\ min\ inf}$	1 mm
$h_{min\ sup}$	40 mm	$h_{min\ inf}$	40 mm
$scala_{ant\ sup}$	2	$scala_{ant\ inf}$	2
$scala_{post\ sup}$	2	$scala_{post\ inf}$	2
$D_{x\ cap, i}$	0 $\forall i$	$D_{y\ cap, i}$	0 $\forall i$
$\frac{d}{h}_{sup}$	0.3	$\frac{d}{h}_{inf}$	0.5
$\frac{d}{h}_{ant}$	1	$\frac{d}{h}_{post}$	1
rib_{AR}	3	$spar_{AR}$	6
rib_{PARAM}	2		
$check_{IN}$	8	$check_{OUT}$	8

Tabella 5.1: Variabili negli esempi di ottimizzazione

Si nota subito, dall'analisi dell'evoluzione del peso totale e delle fraccia massima, che il sistema si sta muovendo verso il suo ottimo; infatti le ultime generazioni del ciclo iterativo tendono verso un valore sempre minore del peso e sempre maggiore, ma entro i vincoli imposti, della freccia di estremità. I valori verso cui tende il peso sono compresi in una banda tra i 1800 ed i 2000 Kg, come si può vedere in figura 5.5, con un minimo (indicato nel diagramma) di 1812 Kg.

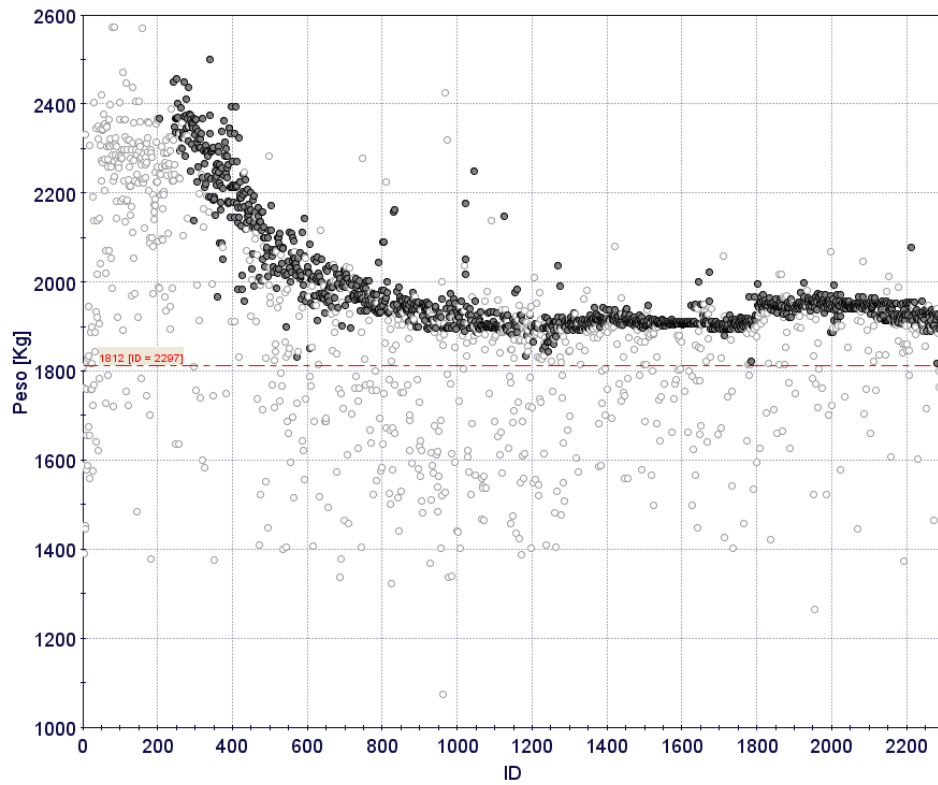
Occorre far presente che il ciclo di ottimizzazione è stato in un primo momento eseguito per 1800 configurazioni con certi valori, in seguito è stato fatto ripartire con valori modificati alla luce da quanto emerso nel corso dell'ottimizzazione e di altre simulazioni svolte in contemporanea. In particolare si è scelto di restringere l'intervallo di variazione del passo dei correnti alla luce della tendenza manifestata dal passo dalle soluzioni ottime (figura 5.1(b)) che, nonostante una certa dispersione dei punti, rimane mediamente nell'intervallo tra 90 mm e 120 mm, e di aumentare lo spessore delle centine da 3.5 a 4 millimetri, visti i risultati di altri calcoli di ottimizzazione (si veda più avanti). Quest'ultima modifica ha comportato un lieve aumento del peso per le prime generazioni calcolate col nuovo valore dello spessore,

come si vede dal salto nell'andamento del diagramma di figura 5.5 per configurazioni oltre la numero 1800, ma nelle ultime ha portato alla soluzione più leggera.

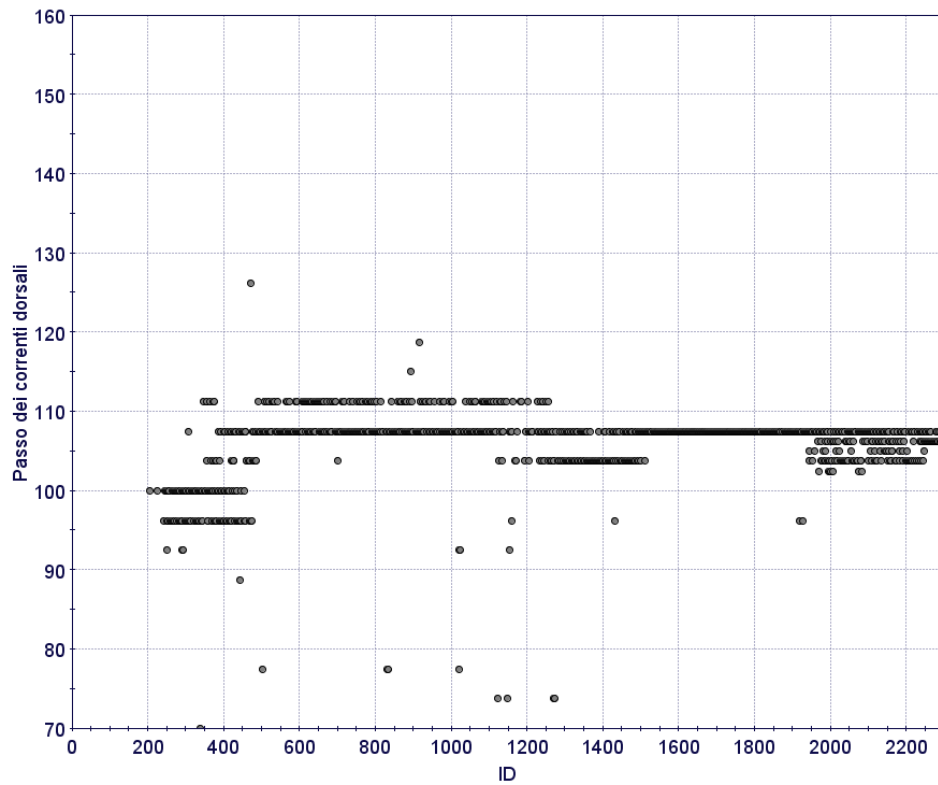
Il valore di freccia massima alla quale tendono le ultime generazioni sta intorno al 7.5%, piuttosto al di sotto del vincolo effettivo del 10%; probabilmente ciò è da imputarsi alla non preponderante importanza di questo vincolo rispetto a quelli sui margini di sicurezza. Ci si può convincere di questo fatto analizzando il diagramma a torta in figura 5.4, nel quale viene indicato, sulla totalità delle configurazioni *unfeasible*, quante lo sono a causa della scarsa rigidità e quante per violazione del vincolo di positività dei margini.

È possibile vedere, in figura 5.2(b) un diagramma peso-freccia al *tip*. Si noti l'andamento iperbolico e come al decrescere del peso diminuisca la rigidità del cassone ed aumenti di conseguenza la freccia, oltre che aumentare il numero di configurazioni che violano qualche vincolo. Una rappresentazione alternativa dello stesso andamento è riportata in figura 5.3, nella quale i punti che identificano le soluzioni vengono sostituiti da bolle il cui raggio è tanto maggiore quanto più è grande il numero progressivo della configurazione e, andando dalle generazioni più vecchie alle più recenti, le bolle diventano più nitide; le soluzioni che rispettano i vincoli sono rappresentate in celeste. Con questa rappresentazione è possibile avere un'informazione del punto di partenza delle prime configurazioni non ottimizzate e di come l'algoritmo le abbia fatte spostare nel corso del ciclo.

Guardando gli andamenti degli spessori, si coglie la tendenza, almeno alla radice, di realizzare un rivestimento più spesso degli irrigidimenti; questo è vero sia per il pannello dorsale che per il ventrale; l'altezza dei correnti dorsali tende al limite superiore imposto, mentre quella dei ventrali, al contrario, tende al limite inferiore, probabilmente perché i correnti superiori hanno una certa influenza sul carico critico di instabilità dei pannelli della baia, che va ad influenzare i margini di sicurezza. I correnti ventrali, invece, non essendo direttamente coinvolti nel calcolo dei margini di sicurezza, vengono trascurati dall'ottimizzatore. Se si fosse imposto un vincolo più realistico sulla resistenza a fatica del pannello ventrale, verosimilmente la tendenza sarebbe stata diversa.

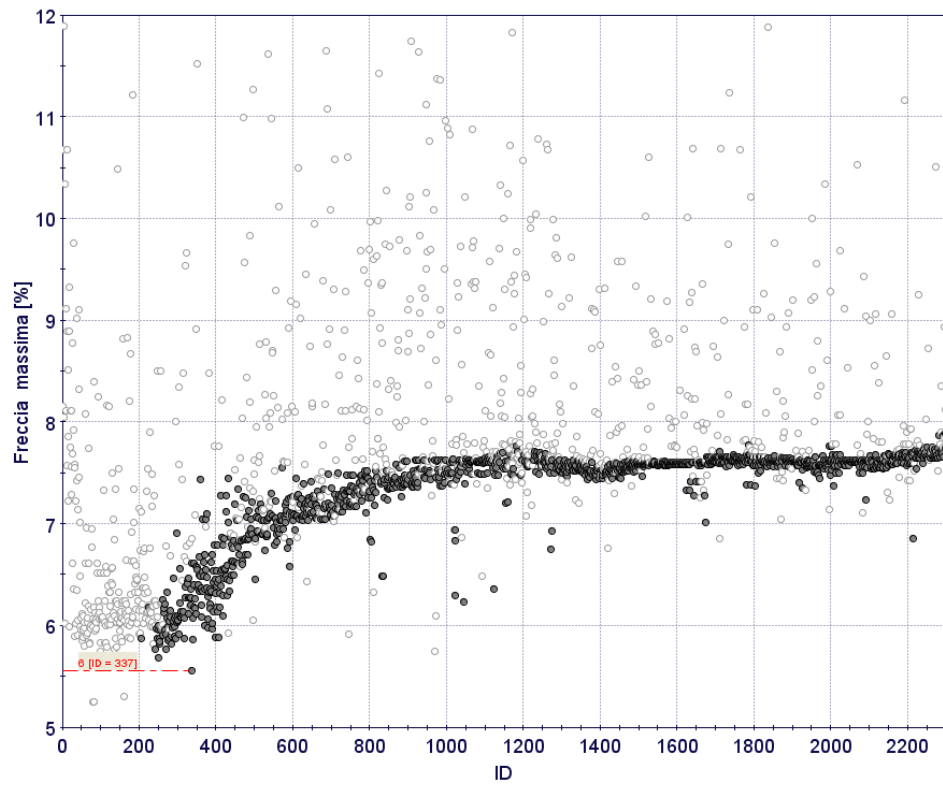


(a)

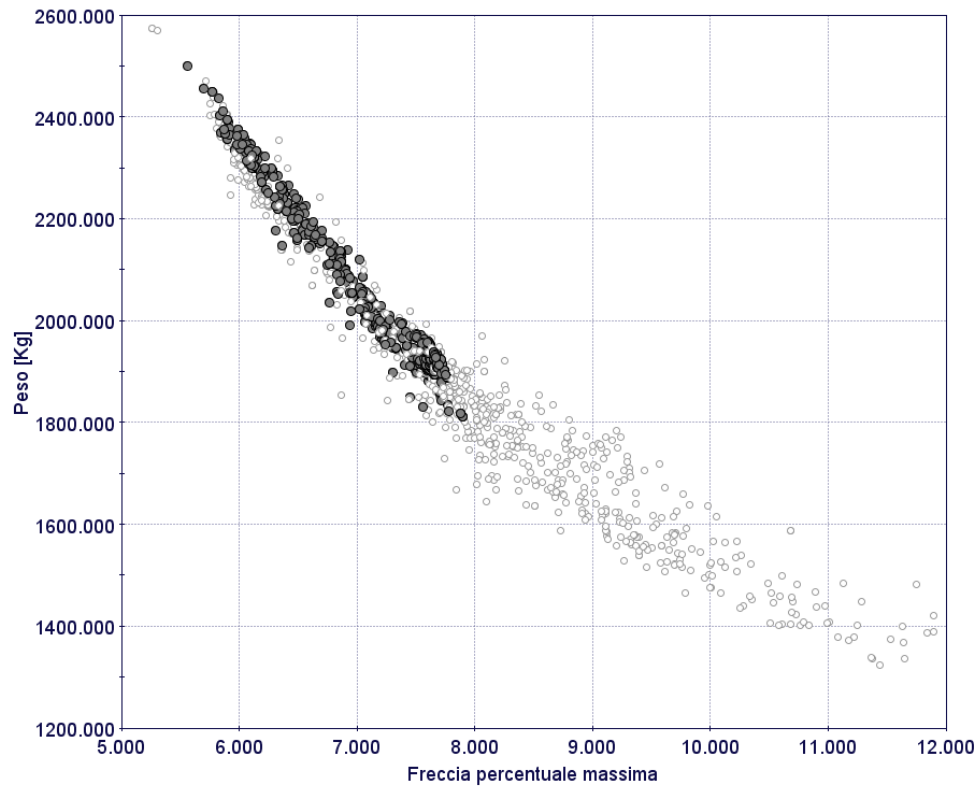


(b)

Figura 5.1: Ottimizzazione del passo dei correnti: peso del cassone (a); passo dei correnti dorsali (b)



(a)



(b)

Figura 5.2: Ottimizzazione del passo dei correnti: freccia massima percentuale (a); *scatter* tra peso e freccia (b)

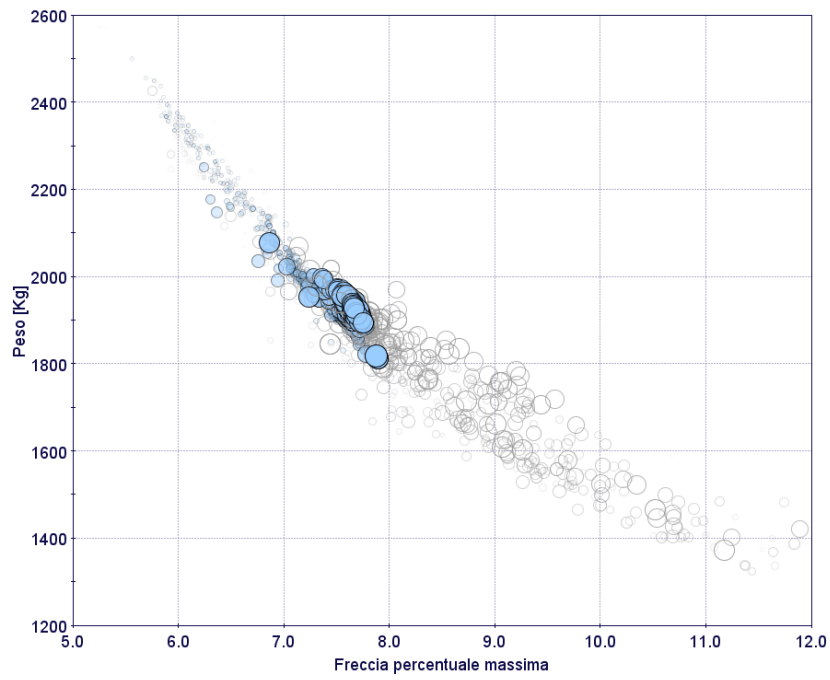


Figura 5.3: *Scatter* tra peso e freccia: rappresentazione con bolle sfumate

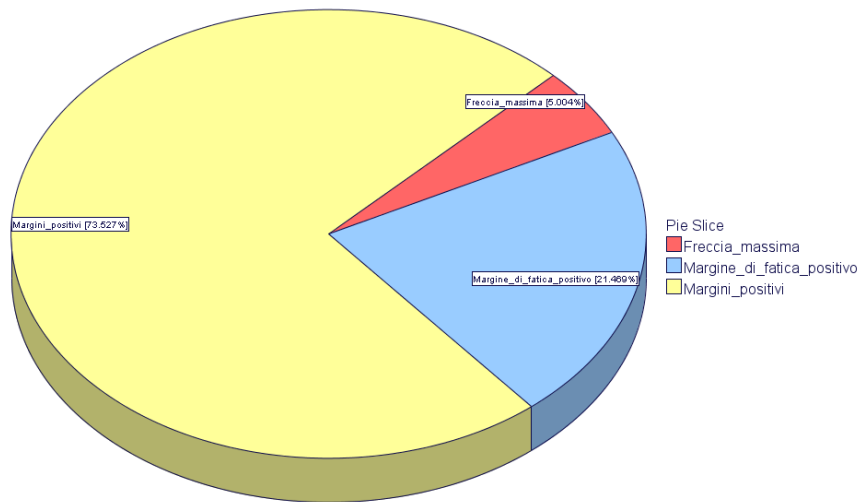


Figura 5.4: Percentuali di violazione dei vincoli

$b_{irr\ sup}$	$70 \div 160mm$ le prime 1800 soluzioni $90 \div 120mm$ le ultime 500
$\Delta b_{irr\ sup}$	$3.75mm$
$b_{irr\ inf}$	$1.3 \cdot b_{irr\ sup}$
L_{in}	350
L_{out}	450
$t_{centine}$	3.5 mm le prime 1800 soluzioni 4 mm le ultime 500
Numero di esemplari di partenza	10 random
Numero di generazioni	180 + 50
Tempo totale di esecuzione	30 ore e 50 minuti
Tempo medio per una configurazione	48.2 secondi

Tabella 5.2: Parametri dell'ottimizzazione dei correnti

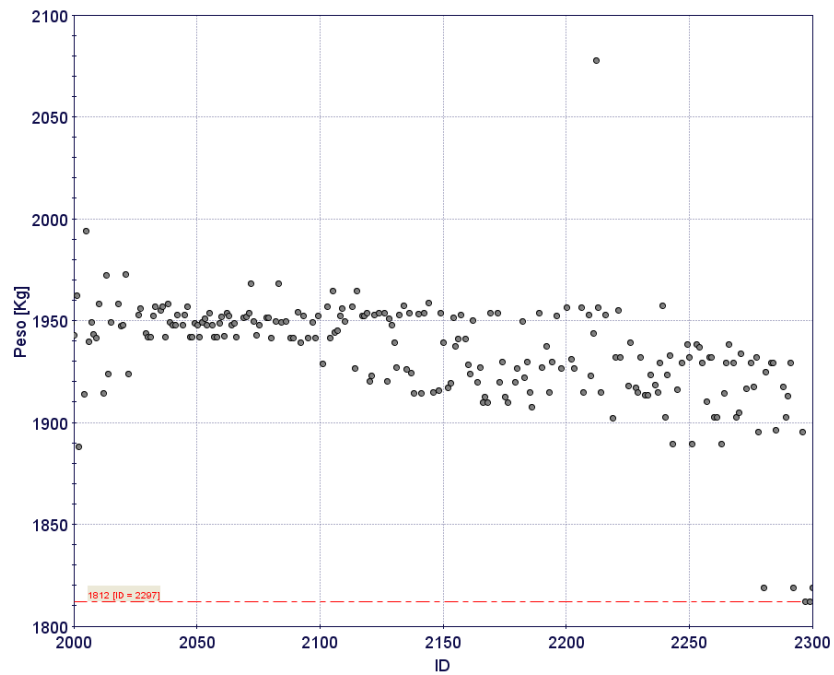


Figura 5.5: Andamento delle peso: dettaglio delle ultime configurazioni

5.2 Confronto: soluzione più leggera e più rigida

	Più leggera (2297)	Più rigida (337)
Peso totale [Kg]	1821	2514
Passo dei correnti [mm]	107.5	70
Freccia massima [%]	7.89	5.55

Tabella 5.3: Confronto tra le soluzioni

Al fine di valutare in dettaglio i risultati prodotti dal processo di ottimizzazione, verranno analizzate la configurazione più leggera, numero 2297, e la più rigida, nonché più pesante, numero 337. Le caratteristiche distintive delle due soluzioni sono riportate nella tabella 5.3; occorre precisare che la soluzione numero 337 è stata scelta perchè dà il valore più basso della freccia massima percentuale, ma non è particolarmente appetibile perchè comunque, essendo apparsa abbastanza presto nel ciclo di ottimizzazione, non risulta ottimizzata al meglio, come emergerà dai risultati. Il vantaggio reale del processo messo a punto è che consente di confrontare soluzioni molto diverse tra di loro, ma che sono entrambe di ottimo rispetto ad un obiettivo imposto. Ovviamente, i tempi di calcolo per arrivare alla situazione ideale auspicata aumenterebbero tangibilmente; alla luce delle osservazioni sul processo svolte nel corso del lavoro del quale la presente è la sintesi si ritiene infatti che, per arrivare a risultati significativi, occorra arrivare a qualche centinaio di generazioni in più, a parità di numero di esemplari di partenza. Inoltre si ritiene che si potrebbero ottenere risultati migliori aumentando i gradi di libertà sulla scelta degli spessori a disposizione dell'ottimizzatore.

Nelle figure, dalla 5.6 alla 5.17, è possibile vedere i risultati, ottenuti con *Patran* e *Matlab*, relativi alle configurazioni in esame.

Nella figura 5.6 è possibile vedere la distribuzione di spessori ed aree per le due configurazioni: si noti che per quasi tutta l'apertura il longherone anteriore è più spesso di quello posteriore e l'andamento costante dello spessore del pannello dorsale di entrambe fino al *kink*. Com'era logico attendersi, la soluzione più pesante ha spessori maggiori in tutta l'apertura e correnti di area maggiore.

Dal confronto degli involucri di margini di sicurezza, in figura 5.7, ci si rende conto che la soluzione leggera è più vicina alla condizione di ottimo, cioè margini di sicurezza molto bassi seppur positivi, anche se migliorabili ulteriormente. La soluzione pesante, invece, ha ovunque in apertura margini piuttosto larghi, segno che non sfrutta bene il materiale a disposizione. I margini relativi alla sollecitazione di taglio sul longherone anteriore, scelta come esmpio perchè la più critica, sono piuttosto vicini all'annularsi per la prima soluzione, perchè evidentemente il processo di ottimizzazione è andato a buon fine. In particolare si

noti come le stazioni in apertura ottimizzate, indicate in figura 5.7(c) abbiano margini molto bassi. Anche le tensioni di taglio nell'anima del longherone sono complessivamente più alte nella soluzione leggera.

Nelle figure 5.8 e 5.9 vengono illustrate le deformazione dei modelli sottoposti alla condizione di volo più gravosa, la manovra a bassa velocità e peso massimo; nelle figure 5.10 e 5.11 sono riportate le tensioni longitudinali sui due pannelli relative a questa condizione.

Nelle figure 5.12 e 5.13 si riportano gli sforzi negli elementi *bar* del modello in corrispondenza di una baia *outboard* in prossimità del *kink*. Anche per i correnti, le tensioni sono maggiori, seppur di poco, per la configurazione leggera. Si noti come le flange si carichino di più ma, in virtù della loro area, sono soggette a sforzi minori; l'andamento delle tensioni nei correnti, variabile in corda ha un massimo spostato verso il longherone anteriore, contrariamente a quanto si sarebbe potuto pensare se si fosse ragionato secondo la teoria elementare.

Nelle figure 5.14 e 5.15, 5.16 e 5.17 infine, vengono riportati gli andamenti dei flussi delle tensioni principali nei due pannelli per entrambe le soluzioni. Si noti ad ulteriore conferma di quanto già detto, come le tensioni nella configurazione rigide siano generalmente più basse sia in trazione sia in compressione.

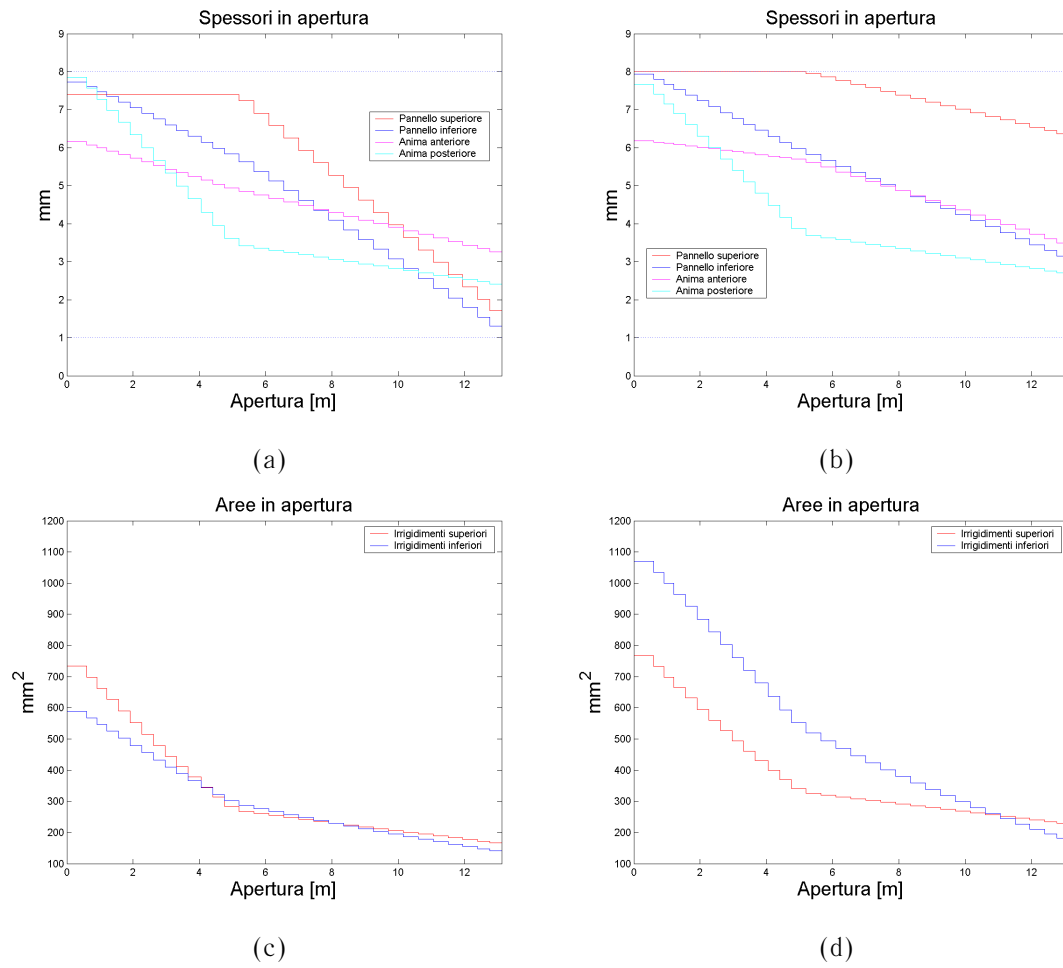


Figura 5.6: Confronto fra caratteristiche lungo l'apertura nelle due configurazioni: spessori (a) ed aree (c) della soluzione più leggera, spessori (b) ed aree (d) della più rigida

5.2. CONFRONTO: SOLUZIONE PIÙ LEGGERA E PIÙ RIGIDA

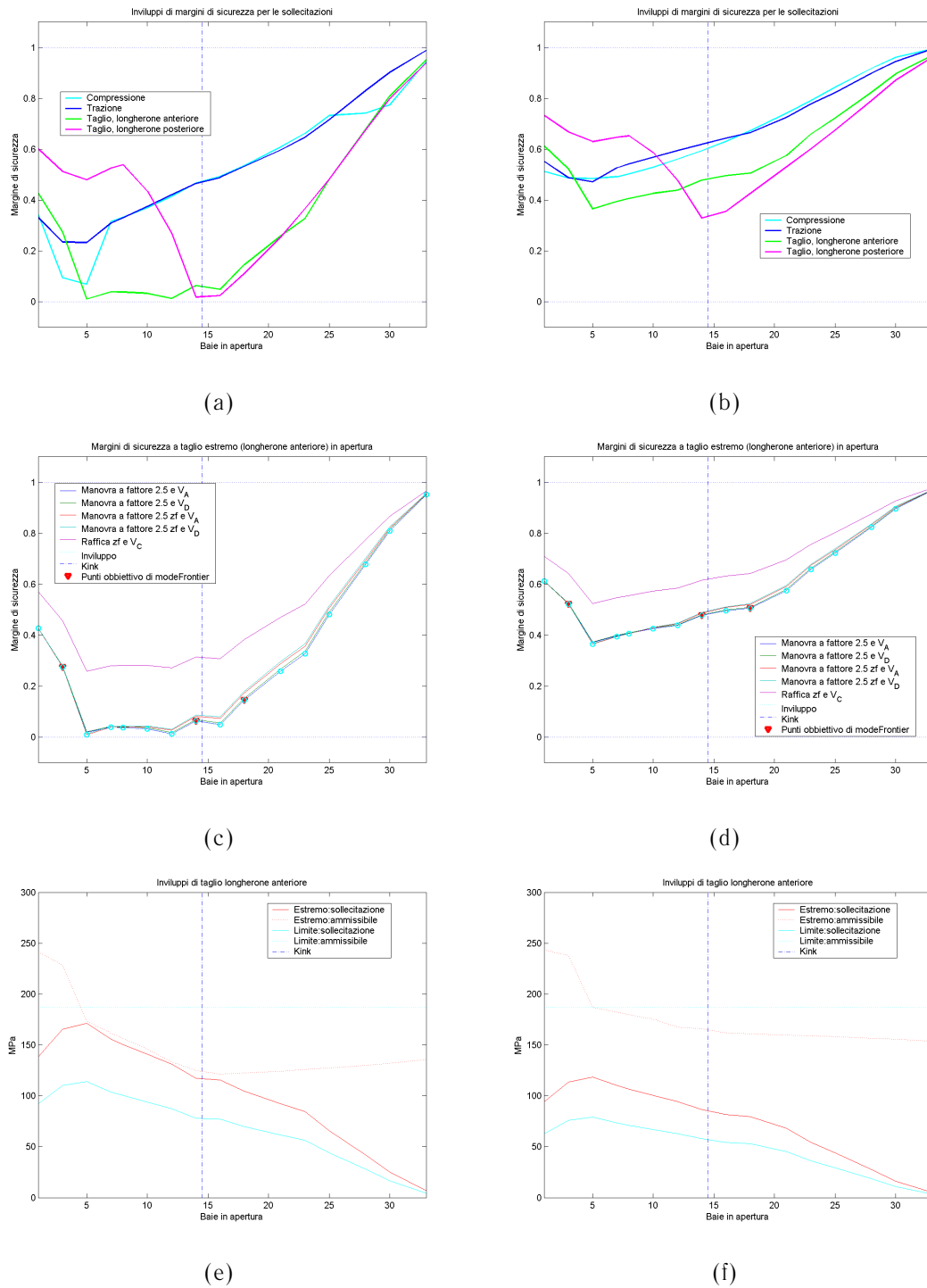


Figura 5.7: Andamento degli involuppi lungo l'apertura per la soluzione più leggera: margini di tutte le sollecitazioni (a); margini di taglio estremo sul longerone anteriore (c); taglio limite ed estremo(e); per la soluzione più rigida: margini di tutte le sollecitazioni (b); margini di taglio estremo sul longerone anteriore (d); taglio limite ed estremo(f)

MD Patran R2.1 04-Jun-09 12:10:16
Deform: Man_FdC_VA A1:Static Subcase, Displacements, Translational, (NON-LAYERED)

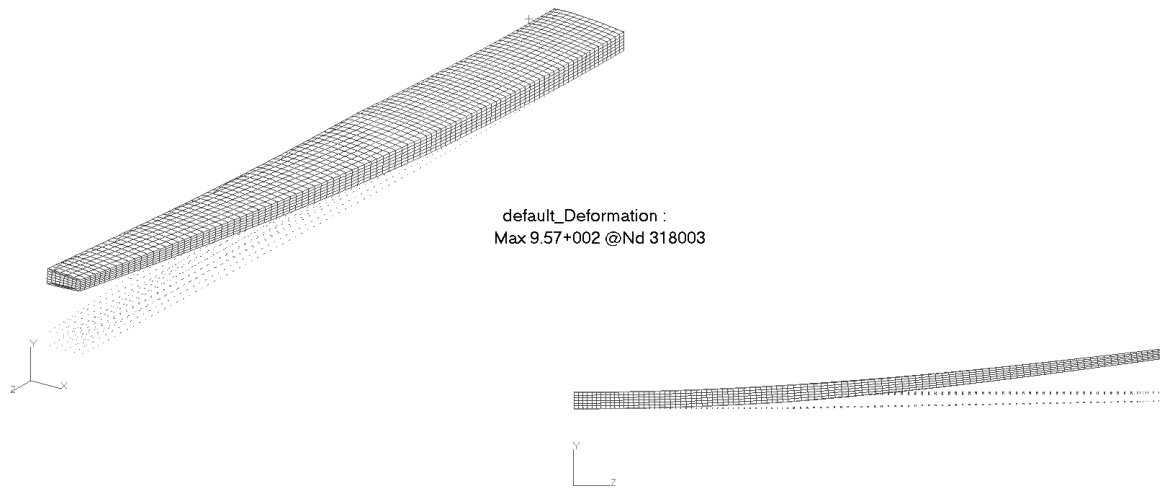


Figura 5.8: Soluzione leggera: deformazione nella condizione di manovra a V_A e peso al decollo, scala della deformazione 1:1

MD Patran R2.1 04-Jun-09 11:48:13
Deform: SC1: A1:Static Subcase, Displacements, Translational, (NON-LAYERED)

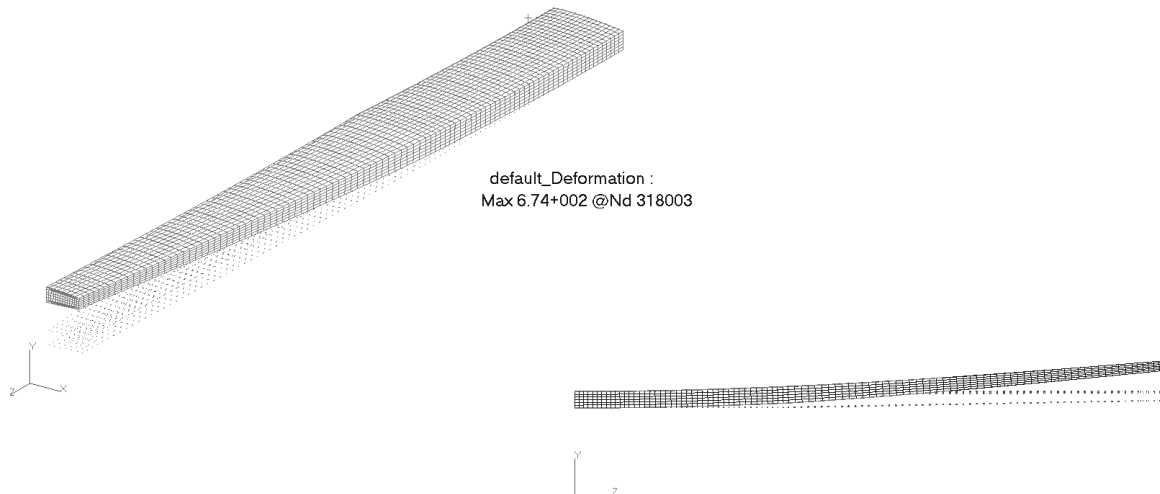
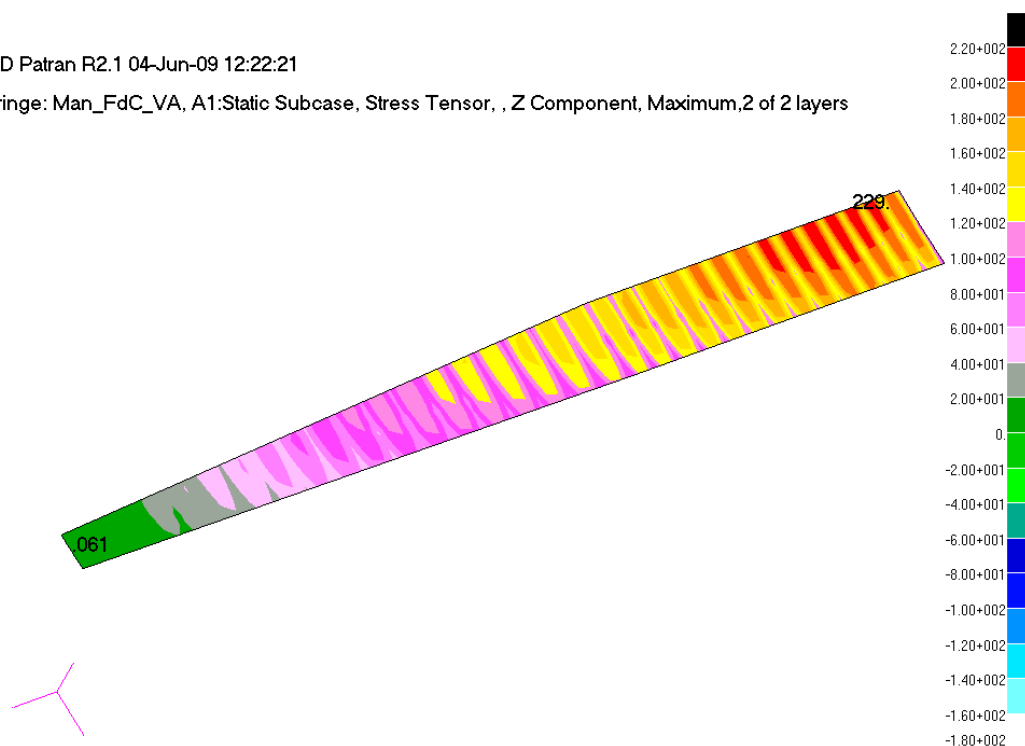


Figura 5.9: Soluzione rigida: deformazione nella condizione di manovra a V_A e peso al decollo, scala della deformazione 1:1

MD Patran R2.1 04-Jun-09 12:22:21

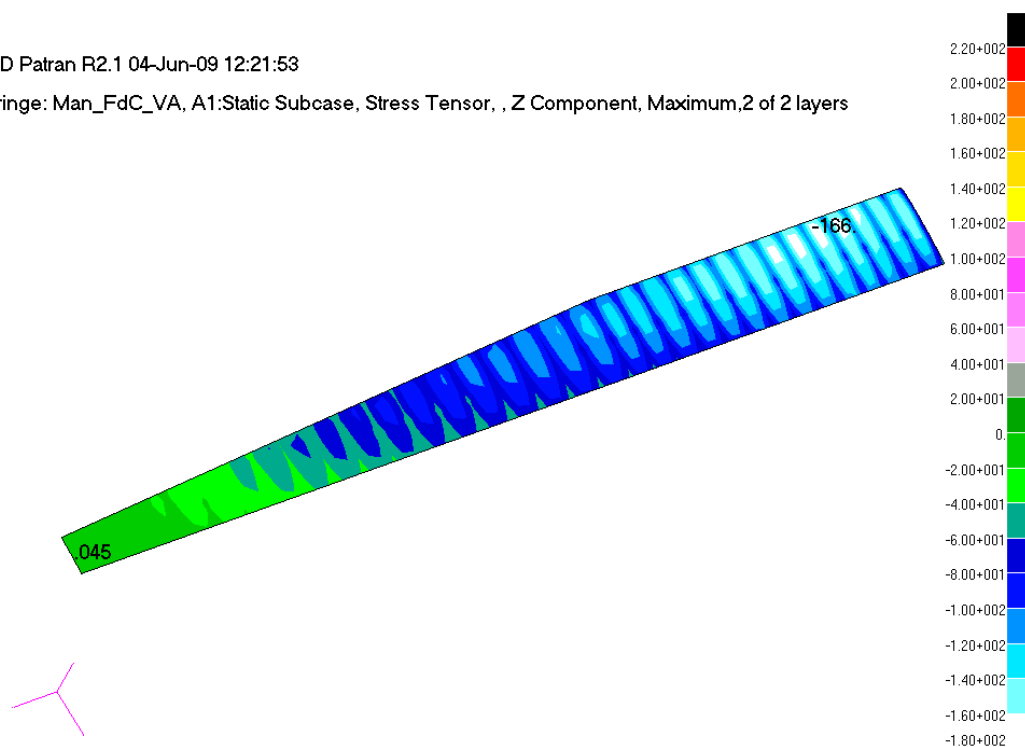
Fringe: Man_FdC_VA, A1:Static Subcase, Stress Tensor, , Z Component, Maximum,2 of 2 layers



(a)

MD Patran R2.1 04-Jun-09 12:21:53

Fringe: Man_FdC_VA, A1:Static Subcase, Stress Tensor, , Z Component, Maximum,2 of 2 layers

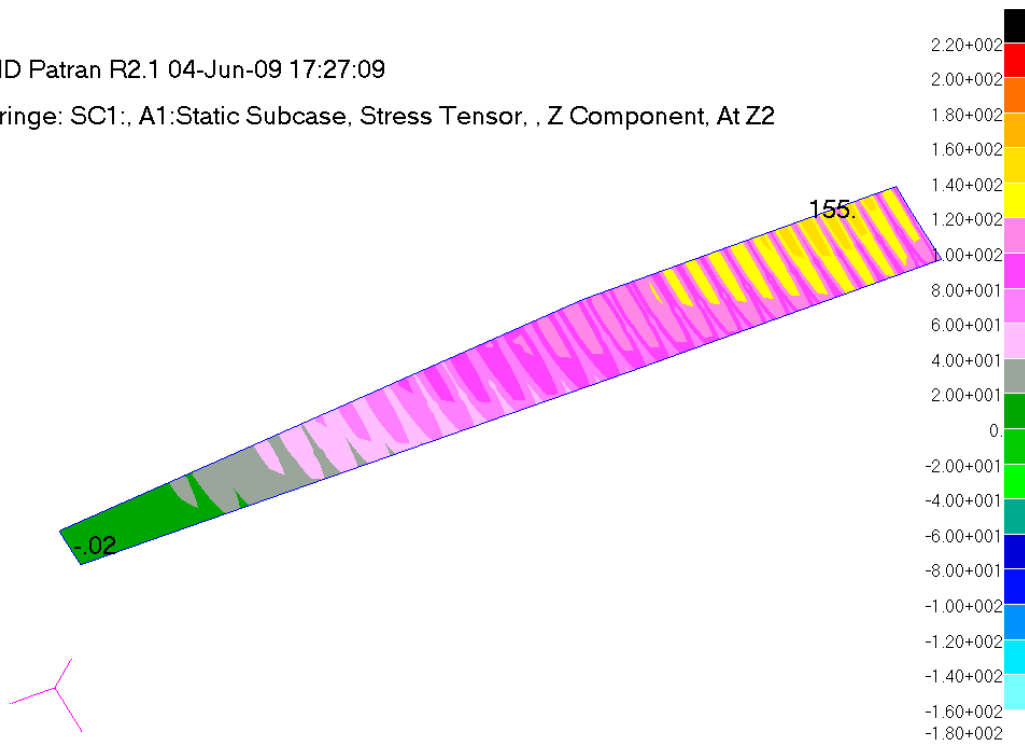


(b)

Figura 5.10: Soluzione leggera: tensioni longitudinali nei pannelli: dorsale (a); ventrale (b)

MD Patran R2.1 04-Jun-09 17:27:09

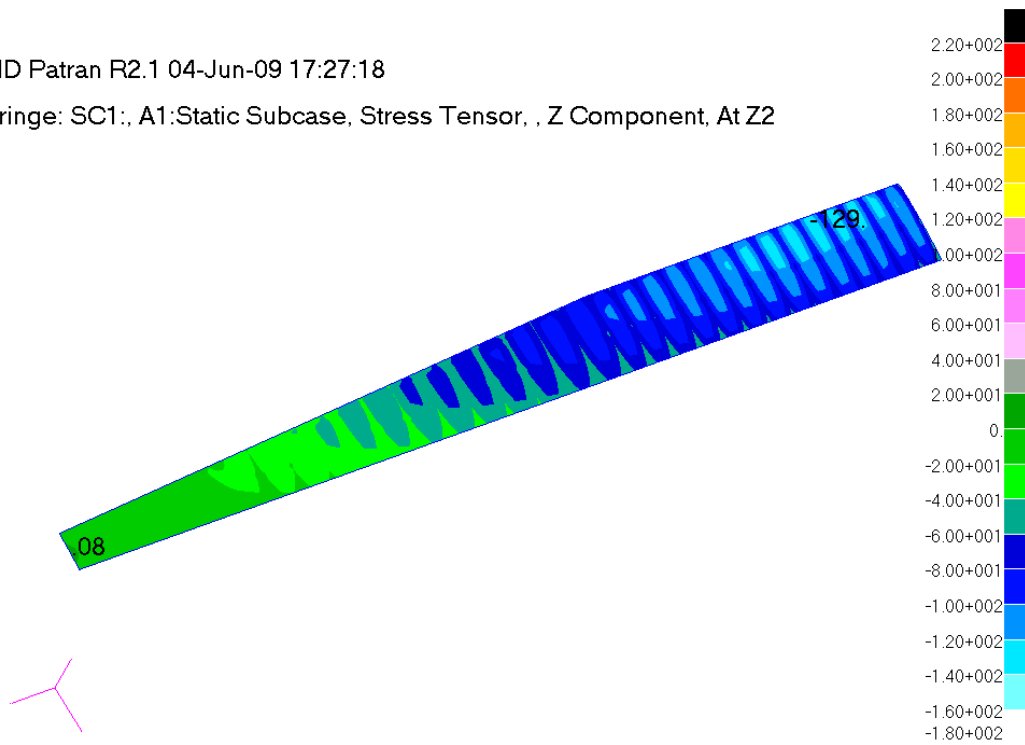
Fringe: SC1:, A1:Static Subcase, Stress Tensor, , Z Component, At Z2



(a)

MD Patran R2.1 04-Jun-09 17:27:18

Fringe: SC1:, A1:Static Subcase, Stress Tensor, , Z Component, At Z2



(b)

Figura 5.11: Soluzione rigida: tensioni longitudinali nei pannelli: dorsale (a); ventrale (b)

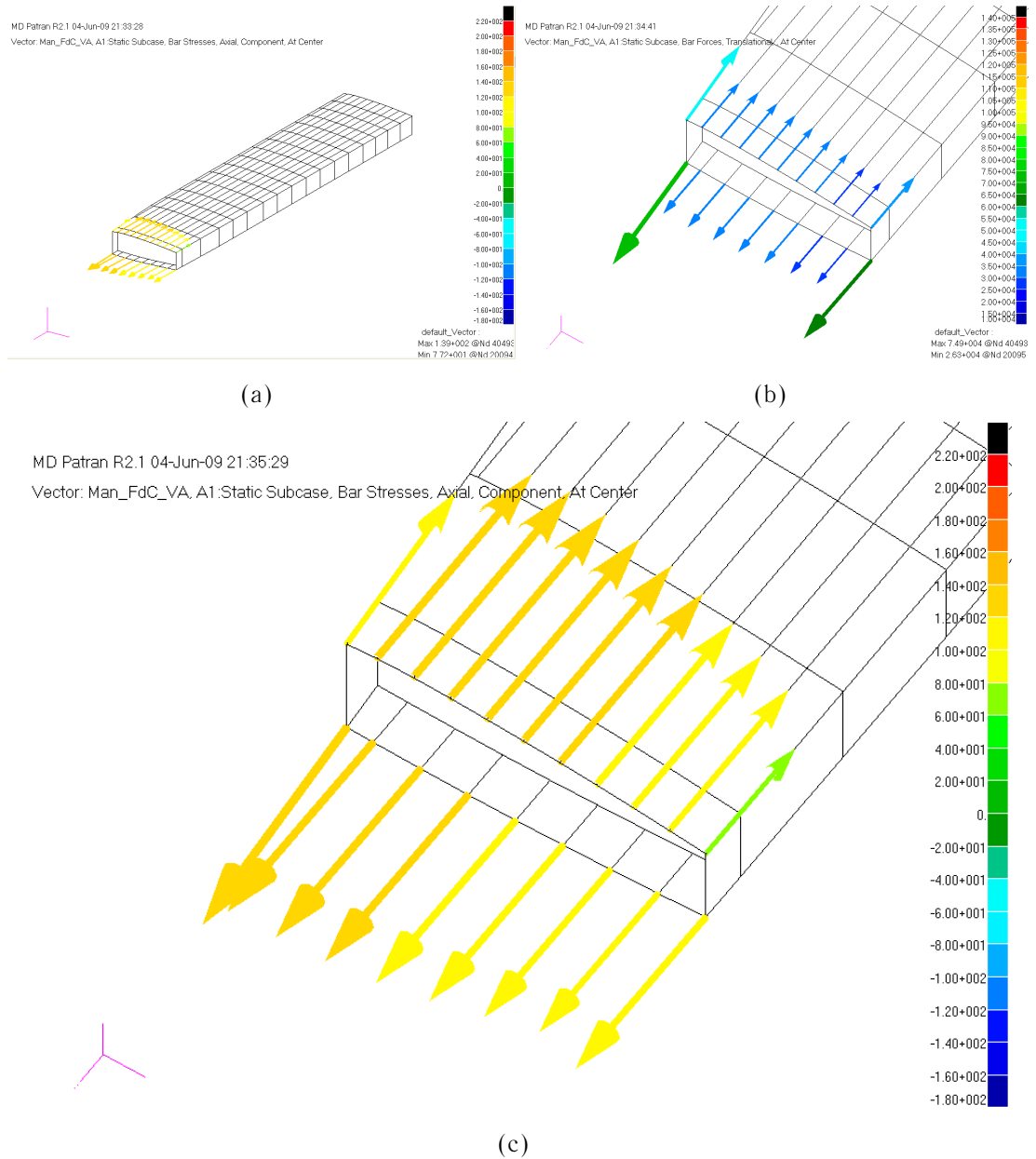


Figura 5.12: Soluzione leggera: tensioni negli elementi longitudinali, vista complessiva (a) e dettaglio (c), forze (b)

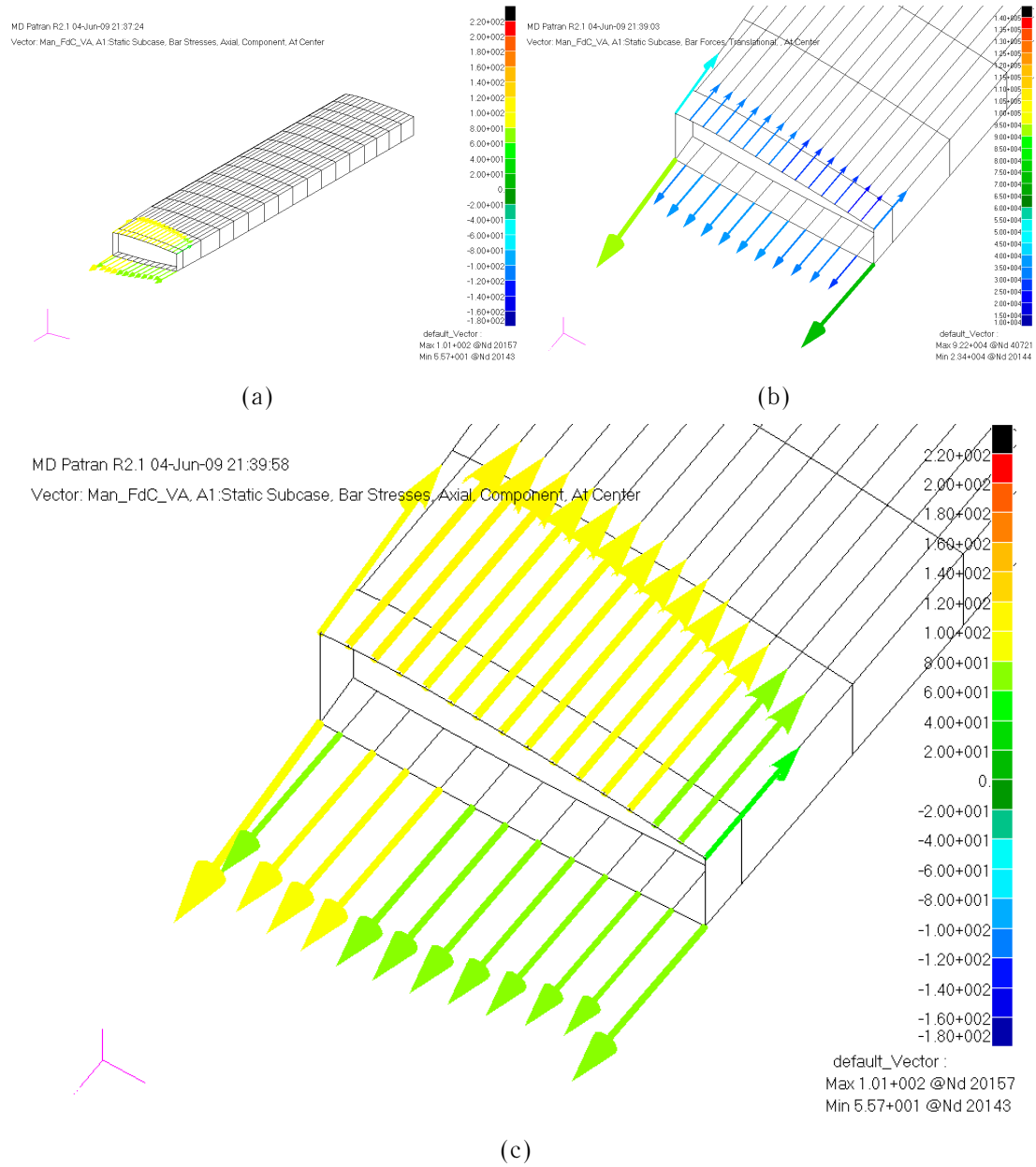
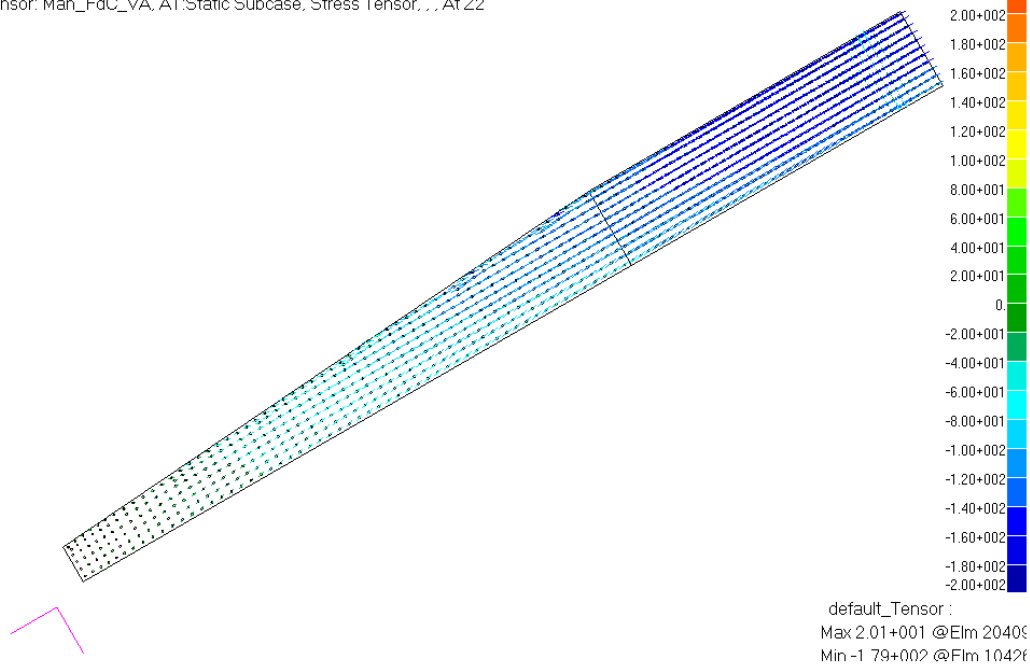


Figura 5.13: Soluzione rigida: tensioni negli elementi longitudinali, vista complessiva (a) e dettaglio (c), forze (b)

5.2. CONFRONTO: SOLUZIONE PIÙ LEGGERA E PIÙ RIGIDA

MD Patran R2.1 04-Jun-09 18:27:58

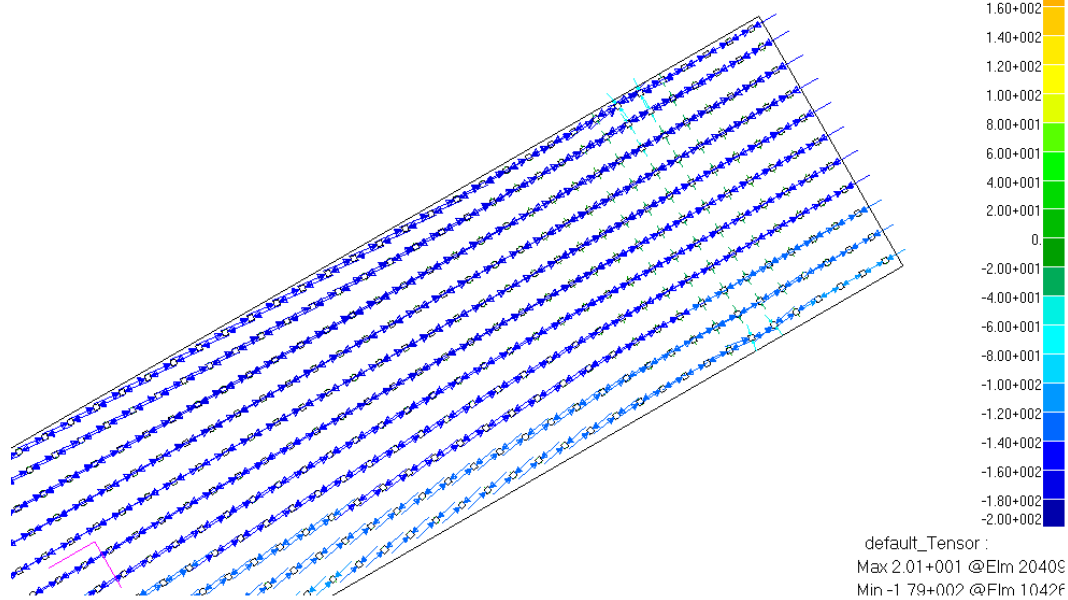
Tensor: Man_FdC_VA, A1:Static Subcase, Stress Tensor, ... At Z2



(a)

MD Patran R2.1 04-Jun-09 18:28:27

Tensor: Man_FdC_VA, A1:Static Subcase, Stress Tensor, ... At Z2

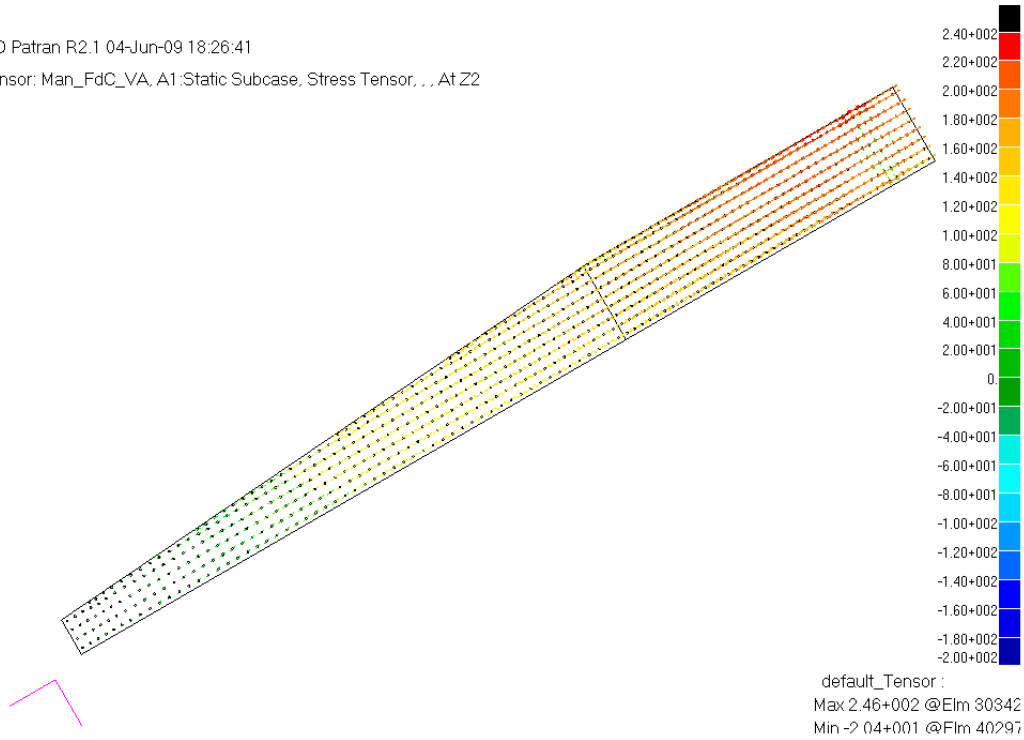


(b)

Figura 5.14: Soluzione leggera: flusso delle tensioni principali nel pannello dorsale (a); dettaglio alla radice (b)

MD Patran R2.1 04-Jun-09 18:26:41

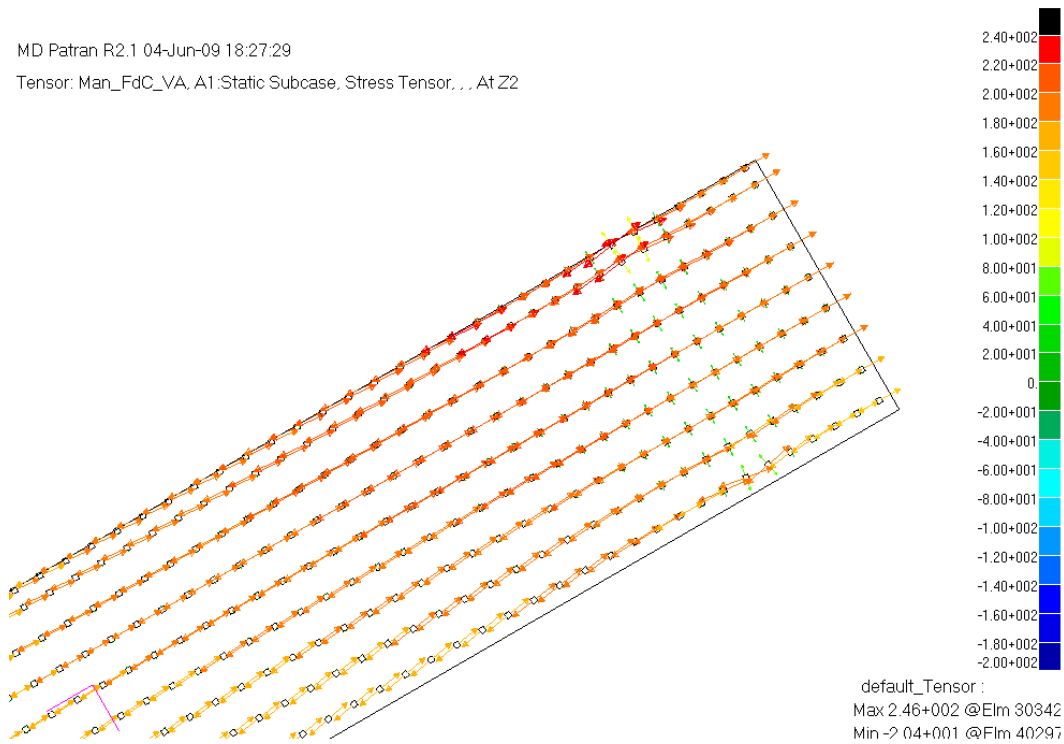
Tensor: Man_FdC_VA, A1:Static Subcase, Stress Tensor, . . . At Z2



(a)

MD Patran R2.1 04-Jun-09 18:27:29

Tensor: Man_FdC_VA, A1:Static Subcase, Stress Tensor, . . . At Z2



(b)

Figura 5.15: Soluzione leggera: flusso delle tensioni principali nel pannello ventrale (a); dettaglio alla radice (b)

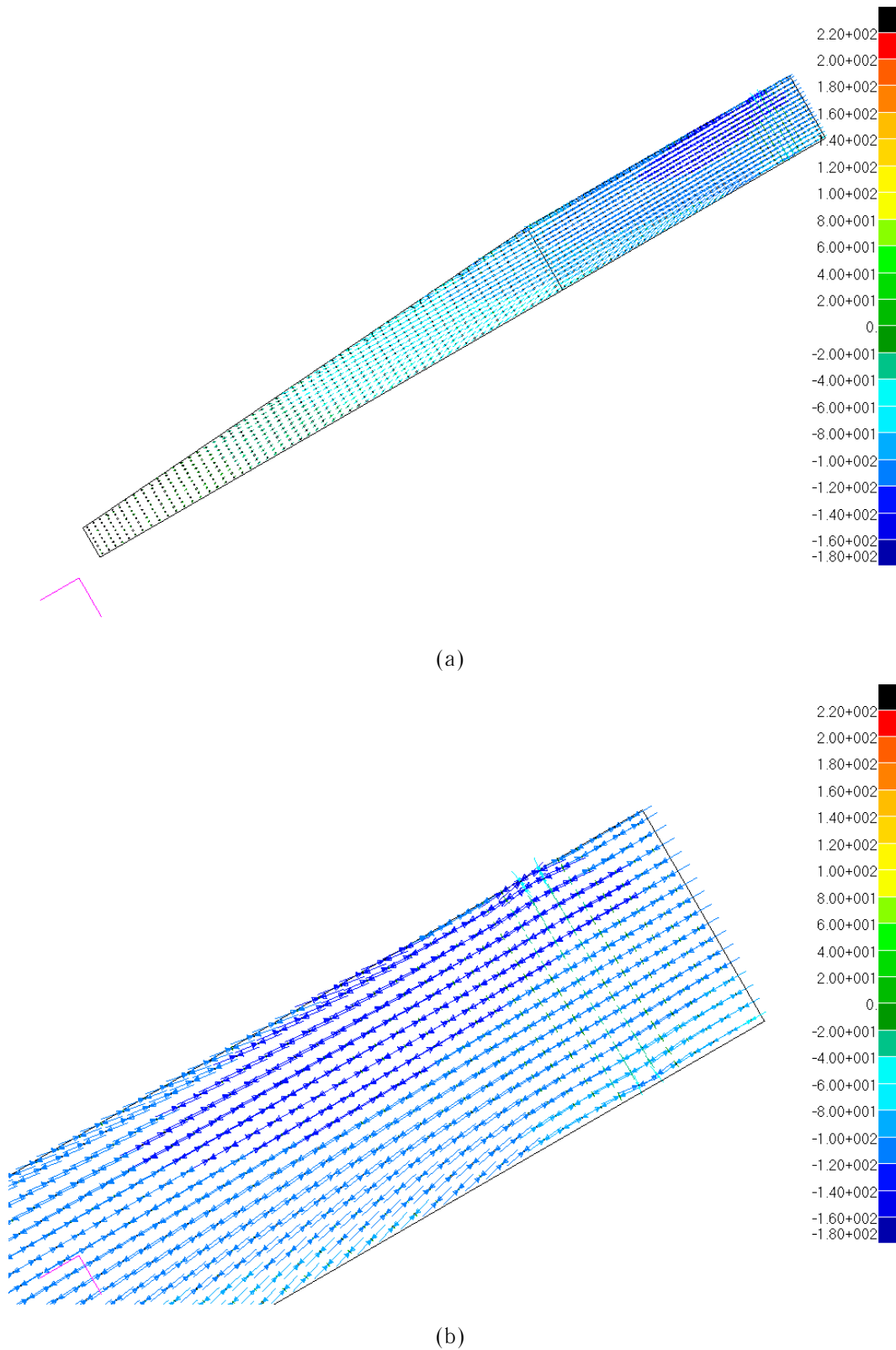
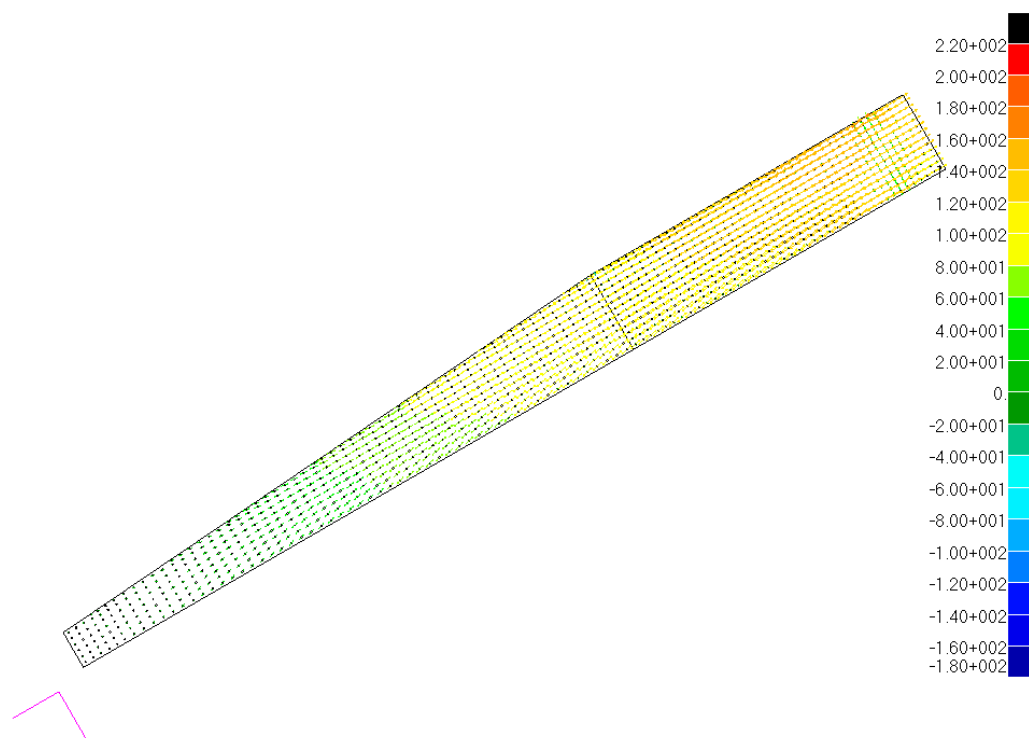
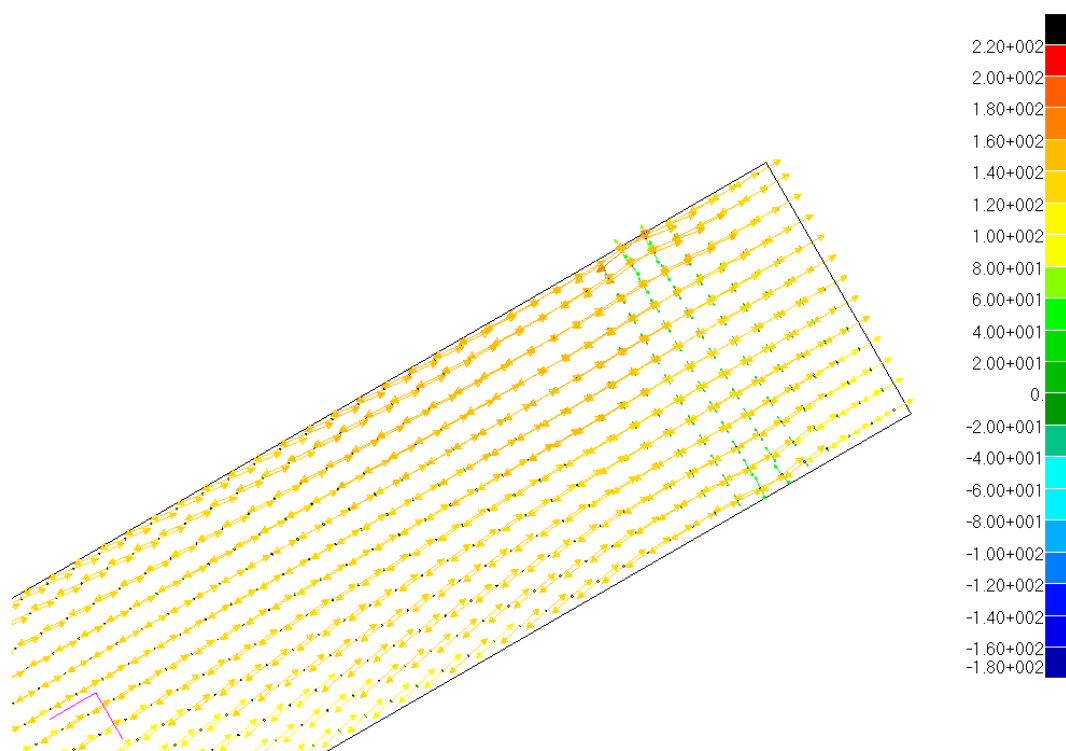


Figura 5.16: Soluzione rigida: flusso delle tensioni principali nel pannello dorsale (a); dettaglio alla radice (b)



(a)



(b)

Figura 5.17: Soluzione rigida: flusso delle tensioni principali nel pannello ventrale (a); dettaglio alla radice (b)

$L_{rib\ in}$	$300 \div 400mm$
$L_{rib\ out}$	$400 \div 550mm$
$\Delta L_{rib\ in}$	$5mm$
$\Delta L_{rib\ out}$	$5mm$
$b_{irr\ sup}$	100 mm
$b_{irr\ inf}$	130 mm
$t_{centine}$	4 mm
Numero di esemplari di partenza	20 random
Numero di generazioni	200
Tempo totale di esecuzione	64 ore e 29 minuti
Tempo medio per una configurazione	58.0 secondi

Tabella 5.4: Parametri dell'ottimizzazione del passo delle centine

5.3 Ottimizzazione preliminare delle centine

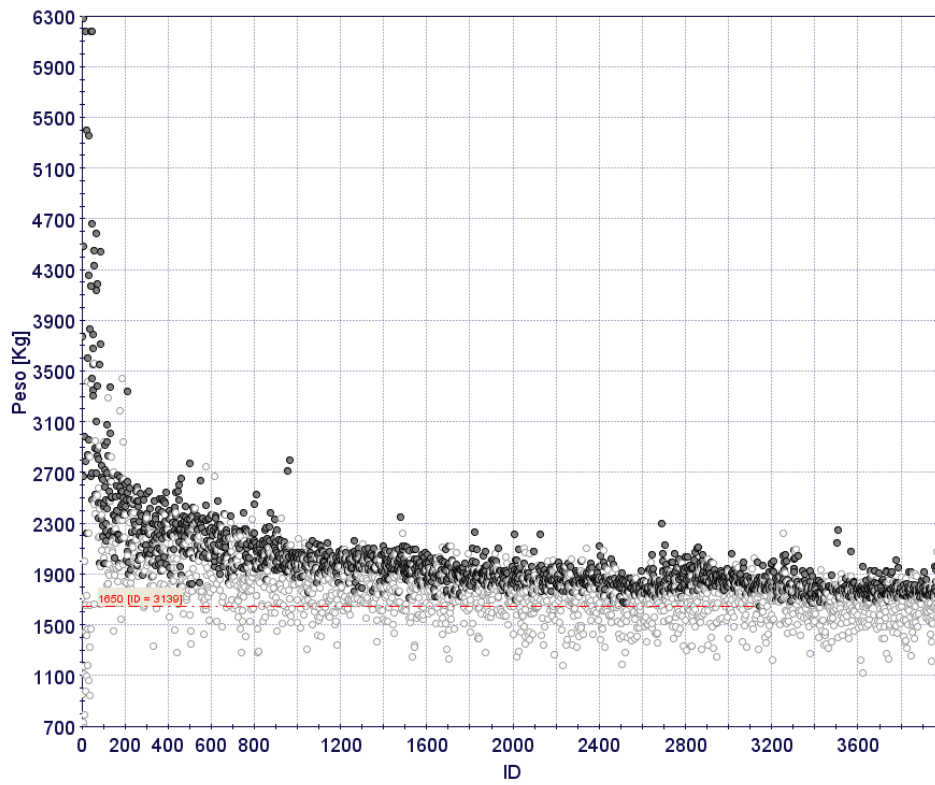
Spaziatura

Nelle figure da 5.18 a 5.19 vengono riportate le storie evolutive delle variabili e lo *scatter* peso-freccia, in maniera analoga a quanto si è fatto nel paragrafo precedente. Rispetto all'esempio dell'ottimizzazione del passo dei correnti, si noti che in questo caso il processo ha vagliato in tutto 4000 configurazioni contro le 2300 del caso precedente. Si tratta quindi di un processo più evoluto, e ne è la prova l'andamento molto pulito delle variabili peso e freccia verso gli asintoti, con miglioramento visibile e tendenza ad ulteriore riduzione del peso. Questo raggiunge un valore molto più basso della soluzione più leggera dell'esempio precedente, anche se per dire il vero non ha senso paragonare due configurazioni diverse, sarebbe molto più logico confrontare il peso medio delle prime generazioni e quello delle ultime per avere un indice più efficace della bontà del processo di ottimizzazione.

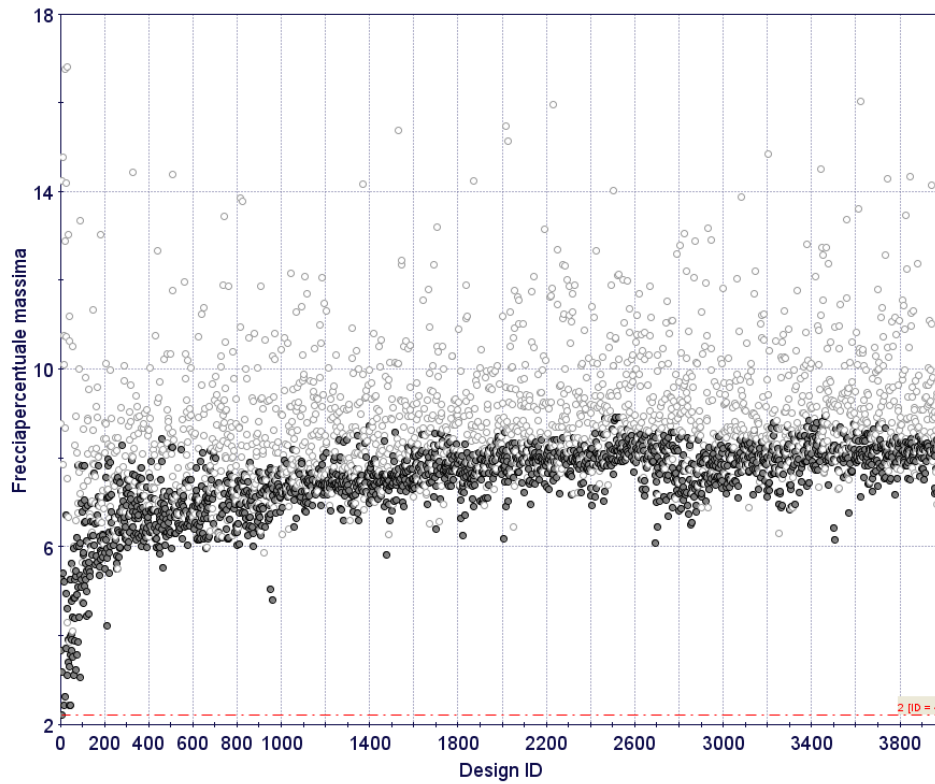
Le spaziature *inboard* ed *outboard*, d'altra parte, non sembrano attestarsi su un valore asintotico quanto piuttosto oscillare tra massimo e minimo dell'intervallo con molto sparpagliamento dei punti. In questi casi è interessante sfruttare gli strumenti di elaborazione statistica del *modeFrontier*; a titolo di esempio si riportano in figura 5.20(a) e 5.20(b) le distribuzioni di densità di probabilità delle due variabili. Si notano, per quanto riguarda la spaziatura delle centine *inboard*, due picchi uno nell'intervallo $345 \div 350mm$ ed uno per $365 \div 370mm$, mentre per il passo *outboard* si vede il massimo nella zona $465 \div 470mm$. Questi dati devono essere valutati in modo critico; infatti non è detto che i picchi si trovino in prossimità dell'ottimo, ma soltanto dei valori che sono capitati più spesso nelle configura-

zioni analizzate dal *modeFrontier*. Questo concetto si visualizza meglio in figura 5.21, nella quale si vede bene come sia possibile evidenziare tutti i *design* che hanno una variabile in un certo intervallo.

I dati ricavati dalle distribuzioni di spaziatore assumono interesse sapendo che i valori del cassone di riferimento sono molto simili a quelli trovati. Comunque, occorrerebbe ripetere il calcolo modificando i limiti dell'intervallo delle spaziatore, per verificare che non si sia trovato questo risultato solo per un caso fortuito.

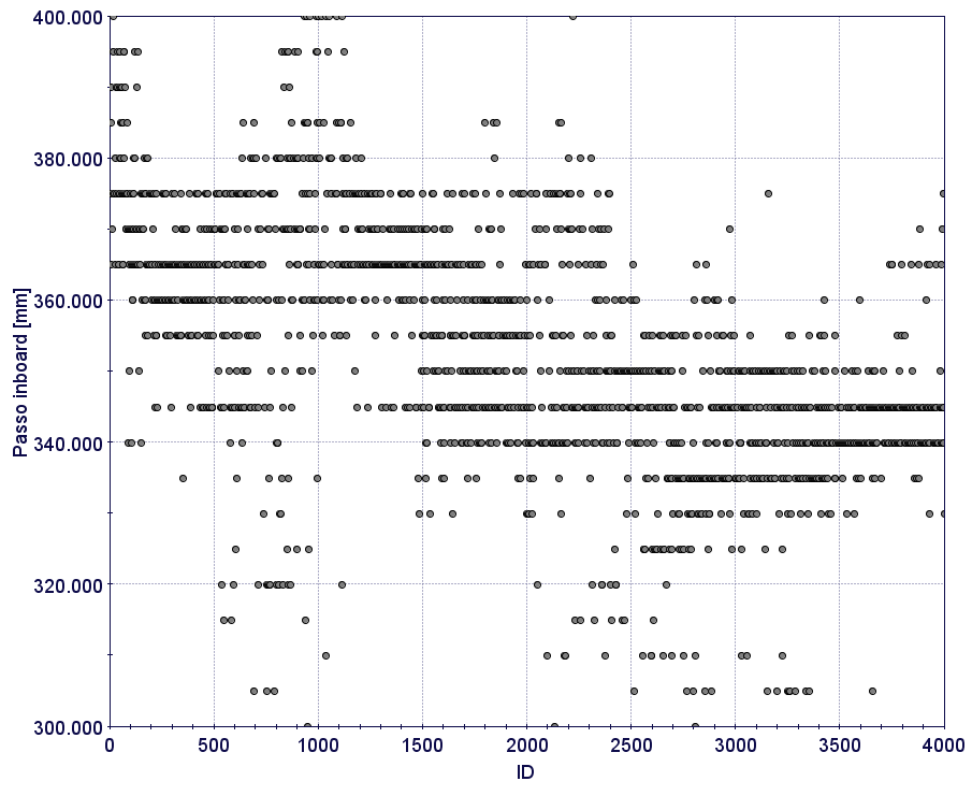


(a)

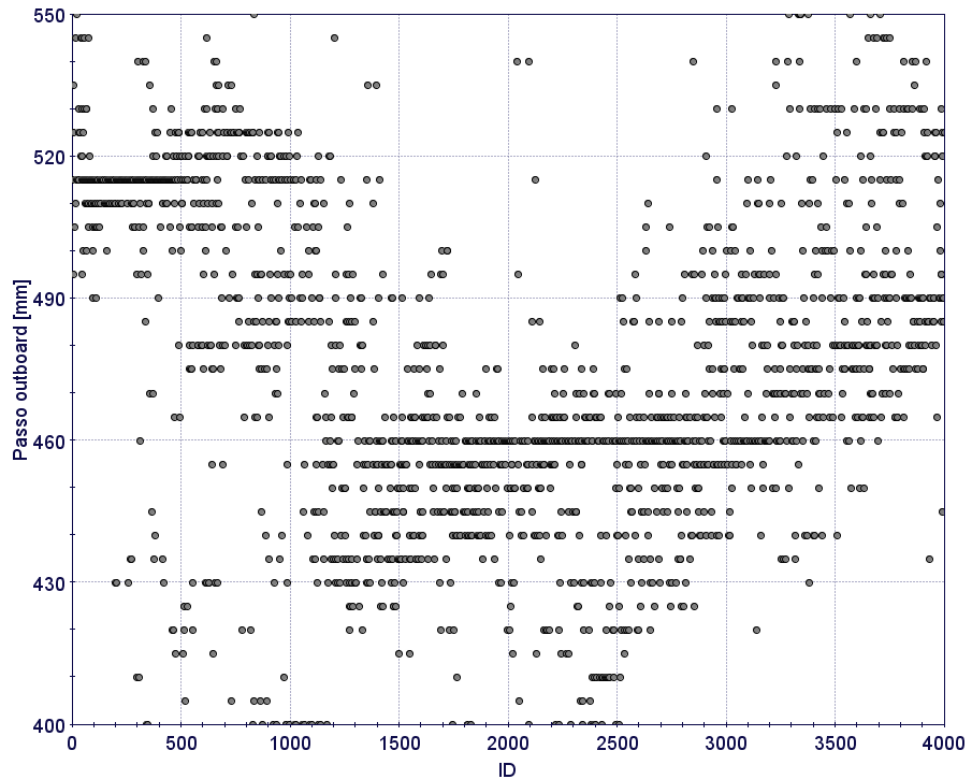


(b)

Figura 5.18: Ottimizzazione del passo delle centine: peso del cassone (a); freccia massima percentuale (b)

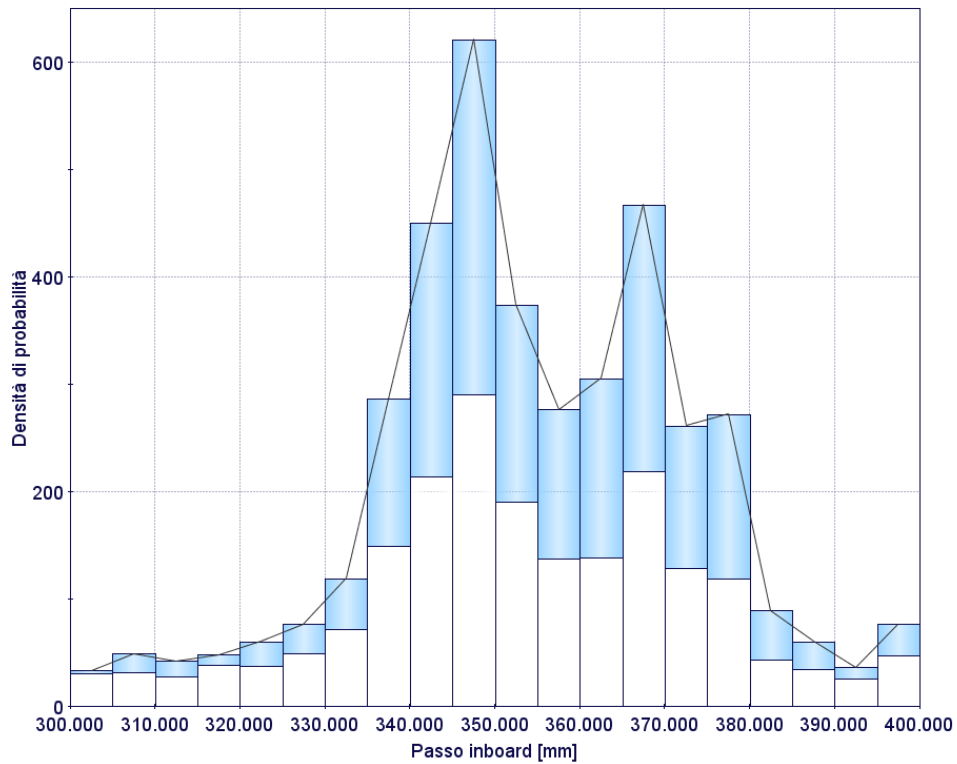


(a)

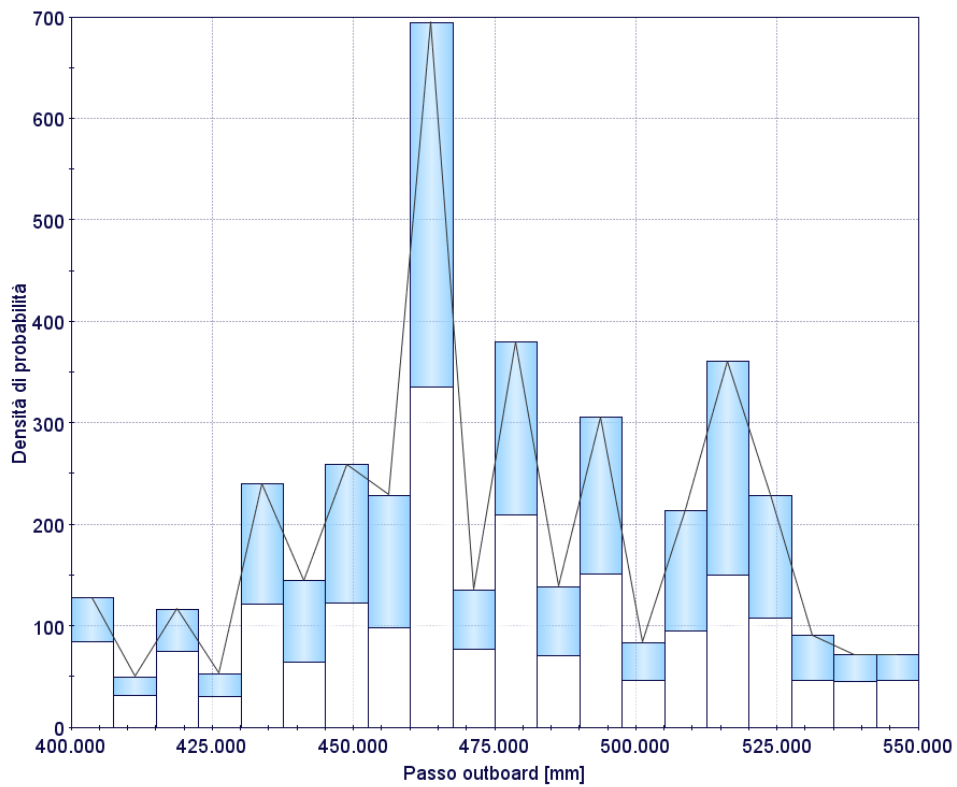


(b)

Figura 5.19: Ottimizzazione del passo delle centine: passo *inboard* (a) ed *outboard* (b)

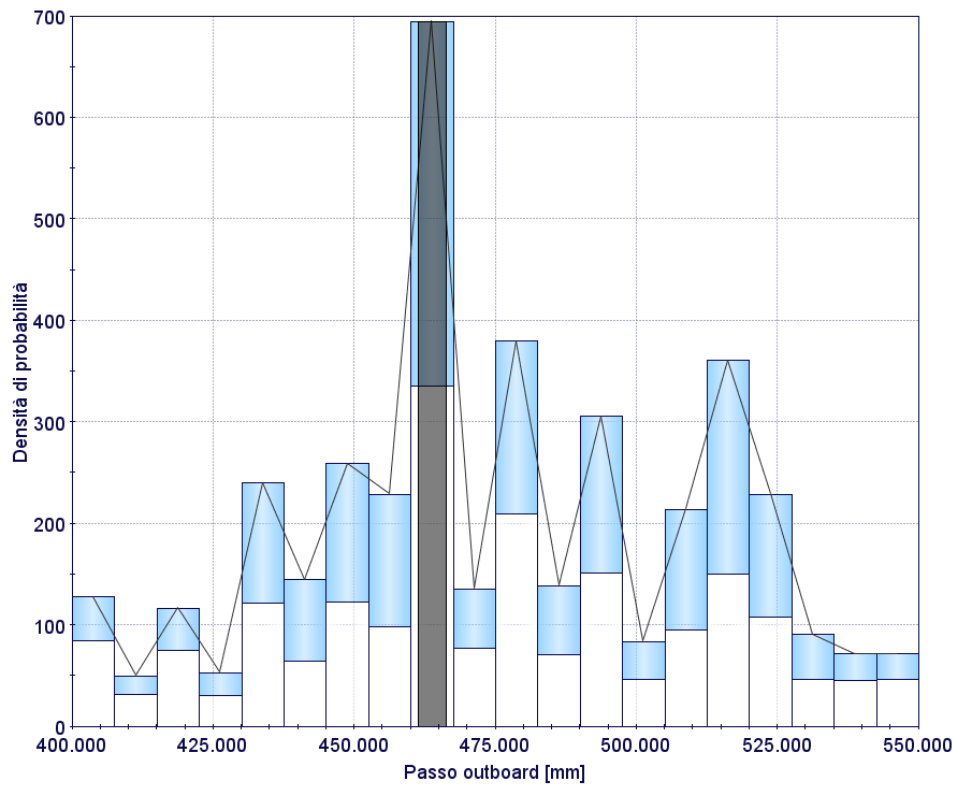


(a)

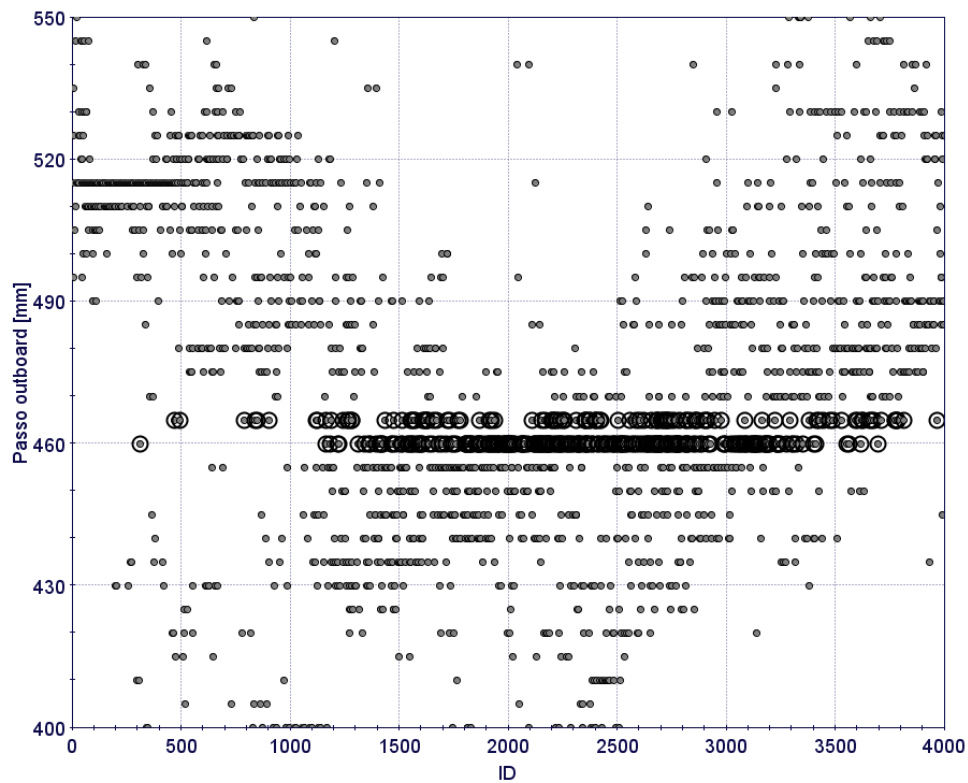


(b)

Figura 5.20: Ottimizzazione del passo delle centine: distribuzione statistica del passo *inboard* (a) ed *outboard* (b)



(a)



(b)

Figura 5.21: Ottimizzazione del passo delle centine: visualizzazione di soluzioni appartenenti ad un certo intervallo di valori

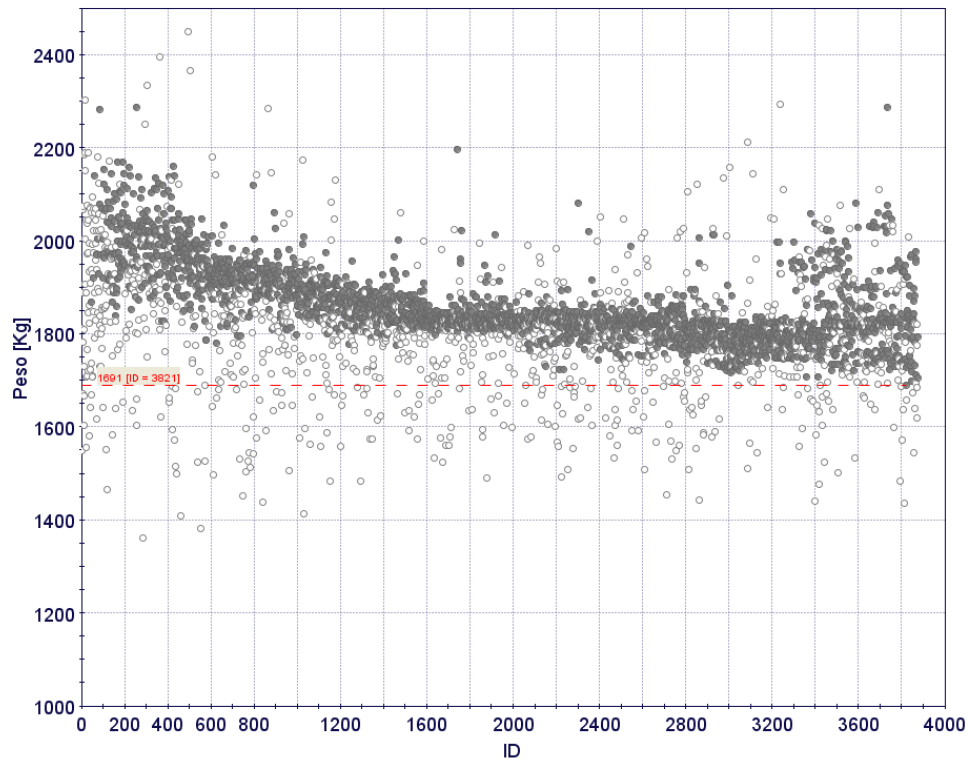
$t_{centine}$	$2 \div 8mm$
$\Delta t_{centine}$	$0.1mm$
$L_{rib\ in}$	$365mm$
$L_{rib\ out}$	$465mm$
$b_{irr\ sup}$	$100\ mm$
$b_{irr\ inf}$	$100\ mm$
<i>Numero di esemplari di partenza</i>	<i>25 random</i>
<i>Numero di generazioni</i>	<i>155</i>
<i>Tempo totale di esecuzione</i>	<i>58 ore e 45 minuti</i>
<i>Tempo medio per una configurazione</i>	<i>54.6 secondi</i>

Tabella 5.5: Parametri dell'ottimizzazione dello spessore delle centine

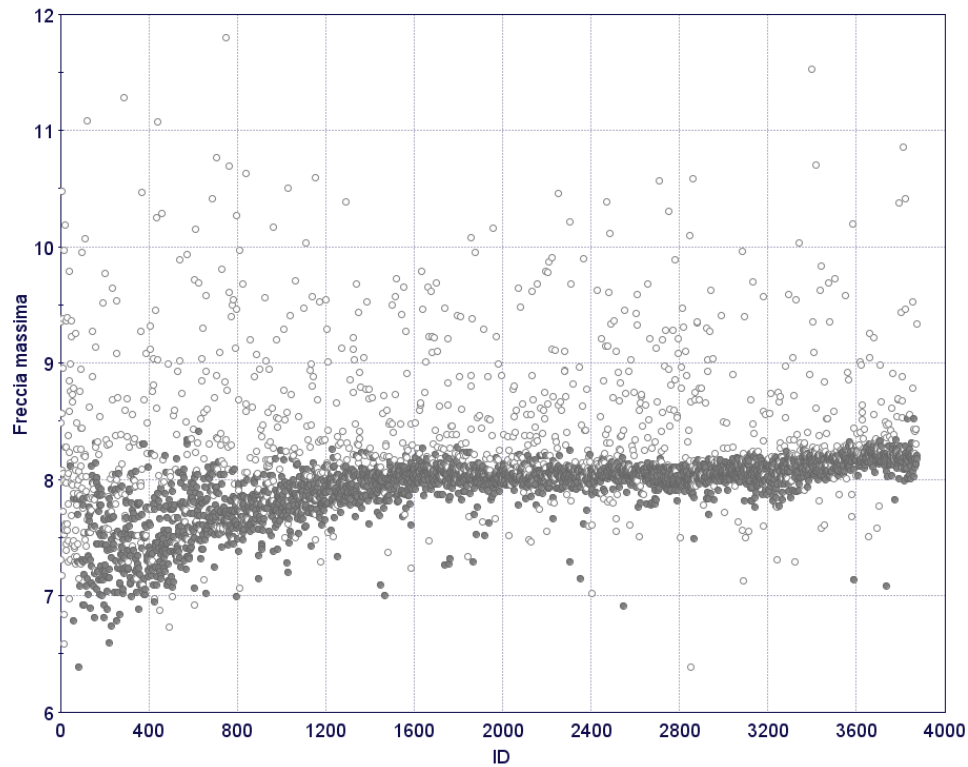
Spessore

L'ultimo esempio che si desidera illustrare riguarda l'ottimizzazione dello spessore delle centine. Come si è già detto, queste hanno dimensione costante lungo tutta l'apertura, e sono rappresentate con piastre senza irrigidimenti. Un calcolo del genere è pertanto estremamente poco rappresentativo del reale comportamento: i setti effettivi saranno rastremati negli spessori, ed avranno montanti verticali. Comunque, anche da un'analisi di questo tipo è possibile ricavare informazioni utili sul problema.

Per questo calcolo ottimizzato i parametri dell'algoritmo genetico sono stati impostati in modo da avere una maggiore probabilità di mutazione e *cross over* sulle generazioni, nell'intento di coprire meglio lo spazio delle soluzioni durante le analisi. Il risultato è un andamento meno regolare, più disperso dei punti soluzione, come si vede nei diagrammi con l'evoluzione di peso e freccia, in figura 5.22. Lo spessore è parso assestarsi sul valore di 3.7 mm intorno alla duemillesima configurazione, invece in seguito è risultato estremamente variabile nel corso dell'ottimizzazione. La soluzione più leggera in assoluto, la numero 3821, pesa 1691 Kg con centine di spessore pari a 2.1 mm; questa soluzione è probabilmente destinata ad essere scartata qualora si mettesse tra i vincoli anche le condizioni di rigidità e resistenza del setto, per ora non contemplate nel modello; la migliore soluzione potrebbe allora essere la 3830, che pesa poco di più (1733 Kg) ed ha uno spessore ottimo delle centine di 3.7 mm.

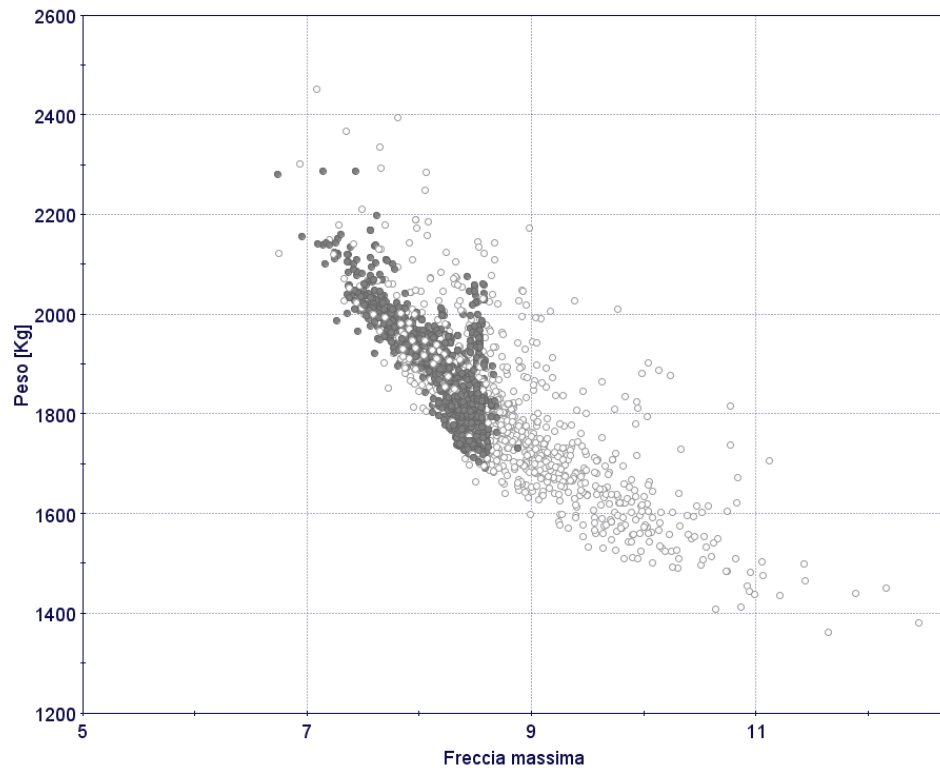


(a)

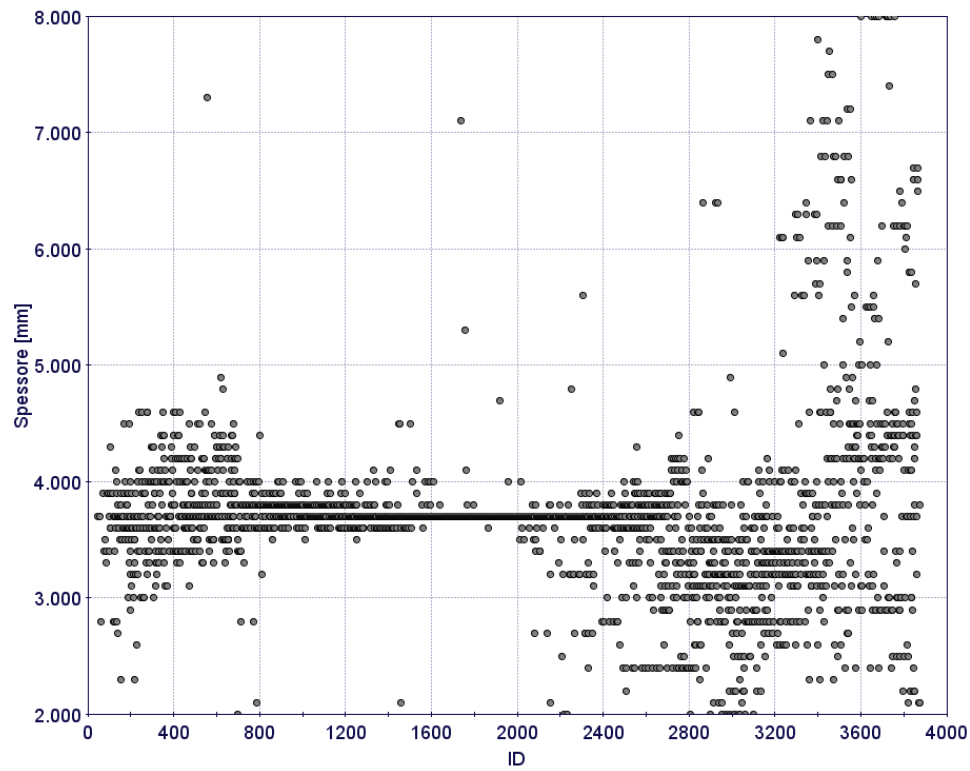


(b)

84
 Figura 5.22: Ottimizzazione dello spessore delle centine: peso del cassone (a); freccia massima percentuale (b)



(a)



(b)

Figura 5.23: Ottimizzazione dello spessore delle centine: *scatter* freccia-peso (a) e andamento degli spessori (b)

CAPITOLO 6

SVILUPPI FUTURI

Il lavoro svolto ed illustrato finora ha portato a realizzare uno strumento di ausilio al progetto di strutture alari. Questo, partendo dalle informazioni preliminari sulla forma in pianta dell'ala e sul tipo di profili, è in grado di fornire un'indicazione importante sulle dimensioni interne dell'oggetto: spaziature, spessori, altezze e tutto quel che serve a definire univocamente la configurazione.

Le informazioni sulla risposta della struttura alle sollecitazioni vengono ricavate dal codice **MD NASTRAN**, che svolge un'analisi con la sequenza di soluzione **SOL101**, che svolge l'analisi statica lineare del modello. Le sequenze di soluzione (vedi (3)) sono un insieme di comandi, nel linguaggio del solutore, che realizzano un tipo di analisi.

L'analisi *non lineare* è il primo passo verso una soluzione più accurata del modello che preveda i comportamenti **plastico**, ovvero oltre il campo di elasticità, e **post-critico**, vale a dire in condizioni di superamento del limite di stabilità della struttura. La risposta della struttura che si comporta in maniera plastica o che abbia superato il carico critico è non più proporzionale alla sollecitazione alla quale viene sottoposta; in queste condizioni svolgere un'analisi lineare porta ad informazioni errate su sforzi e deformazioni, pertanto bisogna ricorrere alla **SOL106** di **NASTRAN**, sequenza di soluzione per l'analisi non lineare. Una panoramica su questo campo dell'analisi strutturale viene riportata nel paragrafo 6.1.

Un altro limite dello strumento creato è nella scelta del materiale: è possibile selezionare soltanto metalli, di fatto precludendo la possibilità di realizzare una struttura in materiale composito, sia esso fibra di carbonio, di vetro, *Glare* o quant'altro. Dato il crescente interesse nel loro impiego specialmente nelle strutture aeronautiche, per le quali è appetibile un costituente che garantisca favorevoli rapporti peso-resistenza, uno sviluppo interessante sarà la possibilità di realizzare il progetto ottimizzato di strutture di questo tipo, le cui problematiche vengono illustrate nel paragrafo 6.2

6.1 Analisi non lineare

6.1.1 Introduzione

Normalmente, nell'affrontare un problema statico viene risolto direttamente il sistema

$$\{\mathbf{P}\} = [\mathbf{K}] \cdot \{\mathbf{u}\} \quad (6.1)$$

nel quale \mathbf{P} ed \mathbf{u} sono il vettore di carichi applicati e degli spostamenti nodali e \mathbf{K} la *matrice di rigidezza del sistema* (da ora in avanti si indicheranno racchiuse con parentesi *quadre* le matrici e con *graffe* i vettori). Nei problemi non lineari quest'ultima dipende anche dagli spostamenti nodali, incogniti, imponendo per la soluzione una strategia iterativa.

La perdita di linearità comporta l'impossibilità di applicare il *principio di sovrapposizione degli effetti*: ogni problema diventa a sé stante e richiede una propria analisi. Le soluzioni non possono essere sommate né scalate e diventa importante la *sequenza* di introduzione dei carichi.

Un'analisi non lineare agli elementi finiti necessita a monte della comprensione del fenomeno fisico studiato, ma anche, e più di quanto accade per i problemi lineari, una maggiore conoscenza dei metodi di calcolo, perché non esiste una strategia risolutiva che vada bene per ogni problema, anzi di solito si ottengono risultati soddisfacenti soltanto dopo averne provate diverse. Inoltre i modelli da sottoporre all'analisi hanno bisogno di una *mesh* più fitta e di una attenta fase di diagnostica, dato che il processo di soluzione è molto più sensibile agli errori di modellazione.

In meccanica strutturale si distinguono fondamentalmente tre tipi di non linearità:

Geometrica Tipica di sistemi caratterizzati da *grandi spostamenti*. Questo comporta che non sia più possibile scrivere le equazioni di equilibrio sulla configurazione indeformata.

Del materiale Se il livello di tensione supera il limite di proporzionalità del materiale costituente, la risposta sarà non lineare a causa della plasticità.

Di contatto Le parti adiacenti di una struttura potrebbero essere fisicamente separate, nel qual caso le forze scambiate all'interfaccia sarebbero funzione dell'area di contatto, che può modificarsi a causa della natura del contatto e dello schiacciamento relativo.

La discussione che segue può essere applicata indifferentemente a qualsiasi tipo di non linearità.

6.1.2 Algoritmi di soluzione

Per capire il modo con cui i codici ad elementi finiti risolvono un problema non lineare, ci si ricondurrà ad un caso monodimensionale: una molla non elastica (figura 6.1(a), tratta da (10)).

Si immagini una molla che abbia costante elastica dipendente dall'elongazione:

$$k = k(u)$$

Posto di non poter ricavare direttamente questa legge (in figura 6.1(b), '*Correct*') ma soltanto valori puntuali della costante elastica, non esistono metodi numerici in grado di risolvere esplicitamente l'equazione; la soluzione avverrà invece con una sequenza di *step* lineari. Questa situazione è la stessa di quando si ha a che fare col sistema a molti gradi di libertà espresso dalla relazione 6.1, con la matrice di rigidezza formata da valori dipendenti dagli spostamenti nodali.

Dunque si procede per passaggi lineari, in ciascuno dei quali viene calcolata la pendenza della curva di rigidezza nel punto attuale (detta rigidezza tangente o *tangent stiffness*):

$$k_t = \frac{dP}{du}$$

dopodiché, dato un incremento di carico ΔP , viene tracciato un segmento rettilineo con pendenza k_t che parte dal punto sul piano (u, P) in cui ci si trova ed arriva in corrispondenza del carico incrementato. In figura 6.1(b) si vede la spezzata '*Calculated*', tracciata secondo il procedimento appena illustrato.

Il problema di questa procedura è che, stanti così le cose, dopo pochi passaggi le due soluzioni divergeranno. Si vede bene che ad ogni *step* si accumula un certo errore, in figura e_{PA} ed e_{PB} , differenza tra valore effettivo e calcolato, che si propaga al procedere del calcolo. Per limitare l'errore, il codice valuta i punti a ed A e confronta la loro differenza con un valore di tolleranza imposto dall'operatore. Se questo è maggiore della tolleranza, il punto soluzione A viene scartato e si riparte da a con un altro segmento rettilineo che arrivi di nuovo in corrispondenza del carico P_1 . Il procedimento prosegue fino a che non si ottiene un valore entro la tolleranza imposta.

La pendenza dei segmenti successivi al primo potrebbe essere pari alla rigidezza tangente attuale, secondo il metodo di **Newton-Raphson** classico, oppure quella di partenza, metodo di **Newton-Raphson** modificato (figura 6.1(c) e 6.1(d) rispettivamente). A prima vista parrebbe poco efficiente usare il metodo modificato, non bisogna però dimenticare che la procedura illustrata deve essere applicata a sistemi a molti gradi di libertà, per i quali potrebbe, in termini di tempi di calcolo, non essere conveniente valutare volta per volta la

rigidezza tangente. D'altro canto usare sempre la stessa rigidezza aumenta il numero di iterazioni da compiere per arrivare alla soluzione; per questo all'utente è consentito definire una strategia intermedia secondo la quale si va avanti nel calcolo con la stessa rigidezza tangente per un certo numero di iterazioni, dopodiché, se non si è arrivati alla convergenza, avviene l'aggiornamento.

Il codice *NASTRAN* prevede anche una serie di algoritmi che migliorano la stabilità e l'efficienza del processo di calcolo. Si rimanda all'appendice D.1 per ulteriori dettagli.

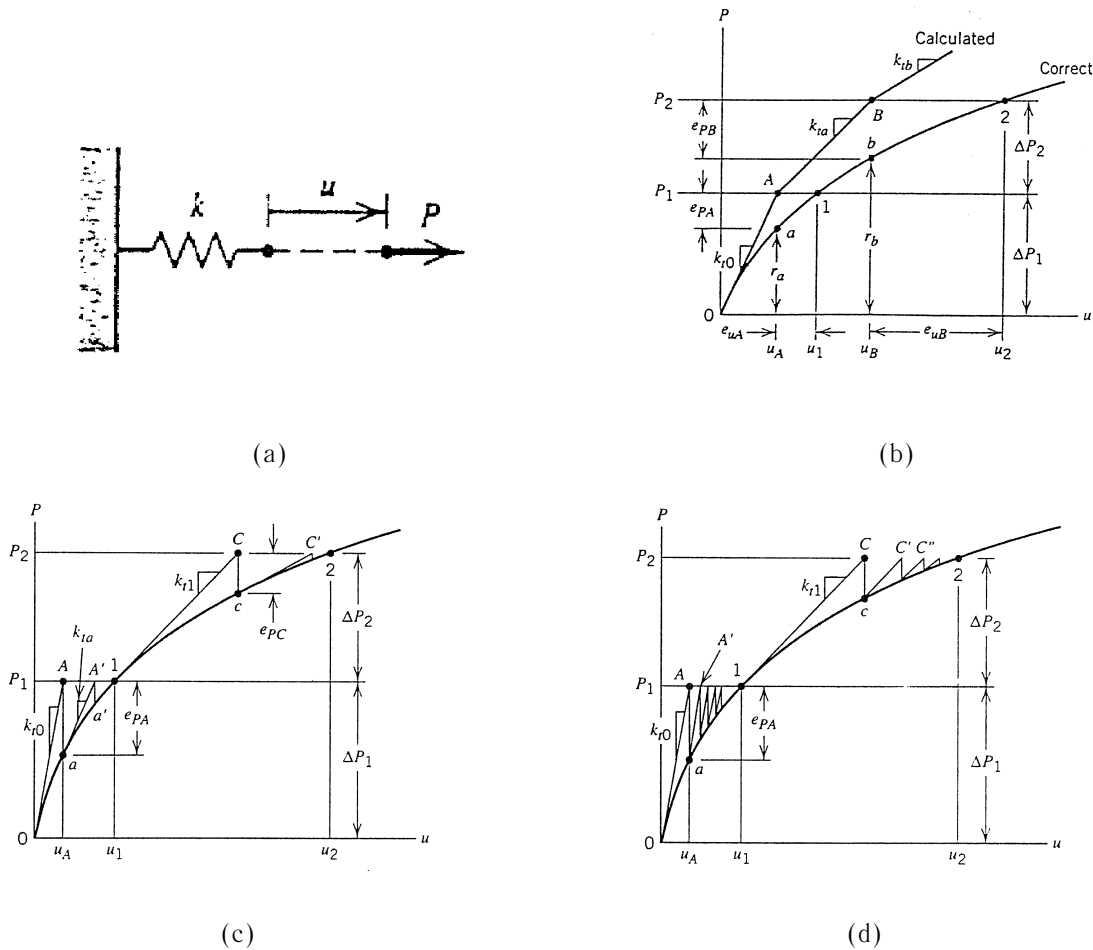


Figura 6.1: Esempio della molla non lineare

La scelta del metodo risolutivo è critica per la buona riuscita dell'analisi. Valutazioni sbagliate infatti portano a costi computazionali inutilmente alti senza miglioramento dell'accuratezza o, peggio ancora, alla mancata convergenza del processo. Nelle figure 6.2 e 6.3 vengono illustrati due tipici esempi di scelta sbagliata della strategia. Nella prima, se si scegliesse il metodo di Newton-Raphson classico si arriverebbe facilmente a convergenza, mentre col metodo modificato il calcolo divergerebbe. Nella seconda, scegliendo il metodo classico ed un certo incremento di carico si potrebbe avere un ciclo intrappolato; in questi

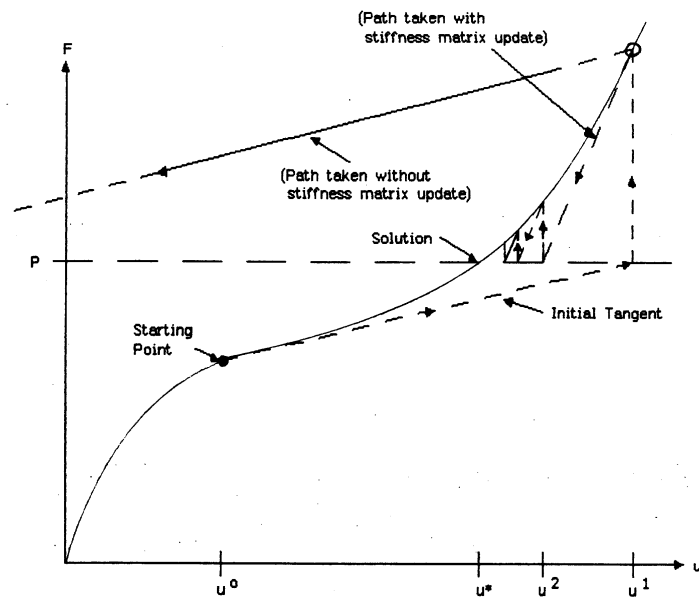


Figura 6.2: Primo esempio di strategia risolutiva sbagliata: materiale incrudente

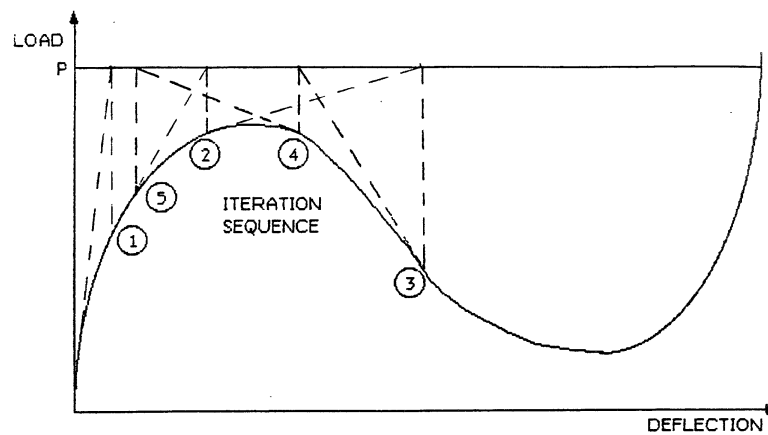


Figura 6.3: Secondo esempio di strategia risolutiva sbagliata: ciclo iterativo intrappolato

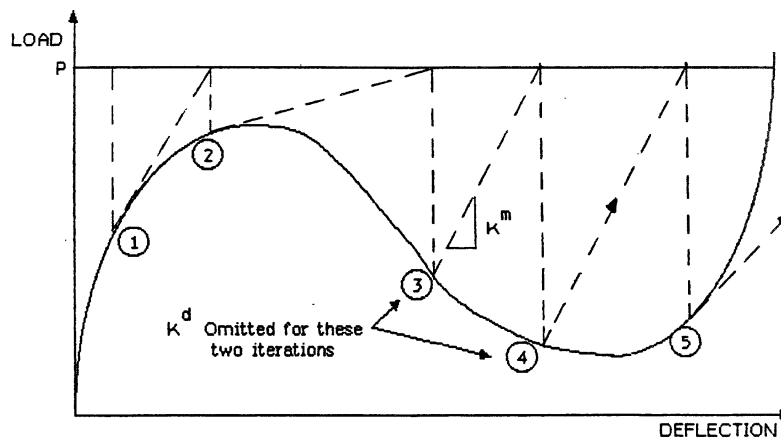


Figura 6.4: Rimedio all'intrappolamento del ciclo iterativo

casi si impone di ignorare la rigidezza tangente qualora diventasse negativa, dimodoché il calcolo possa procedere (figura 6.4).

Per i sistemi che presentano un massimo ed un minimo della curva sforzi-deformazioni, soggetti cioè ad instabilità a scatto (*snap-through*), i metodi di Newton-Raphson e loro derivati non funzionano efficientemente. Infatti questi algoritmi presuppongono di imporre ad ogni punto calcolato un incremento del carico nello *step* successivo. Questo li rende inadatti a trovare le concavità verso l'alto della curva. In questi casi è conveniente sfruttare i metodi basati sulla lunghezza d'arco (*arc-length*), in cui non si procede per incrementi di carico, ma una funzione del carico e degli spostamenti tale per cui si traccia un arco di curva, da cui deriva il nome del metodo, centrato sul punto-soluzione allo *step* attuale (figura 6.5). Poiché l'algoritmo esplora anche i punti al di sotto del precedente, è in grado di individuare cambi di pendenza e concavità verso l'alto sulla curva. Tra i metodi di questa famiglia, il più usato è quello di Crisfield.

In figura 6.6 è presentato un confronto tra metodo di Newton-Raphson e metodo di Crisfield per la soluzione di un problema di *snap-through*. Sia in fase di carico che di scarico del modello, Newton-Raphson arriva fino al massimo o minimo della curva e poi 'salta' ad uno spostamento maggiore o minore senza rendere conto del cambio di pendenza. Il metodo di Crisfield, invece, rappresenta correttamente l'andamento della soluzione.

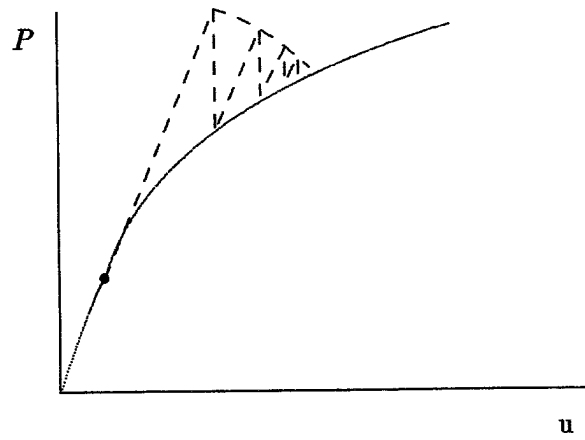


Figura 6.5: Esempio di uso del metodo della lunghezza d'arco

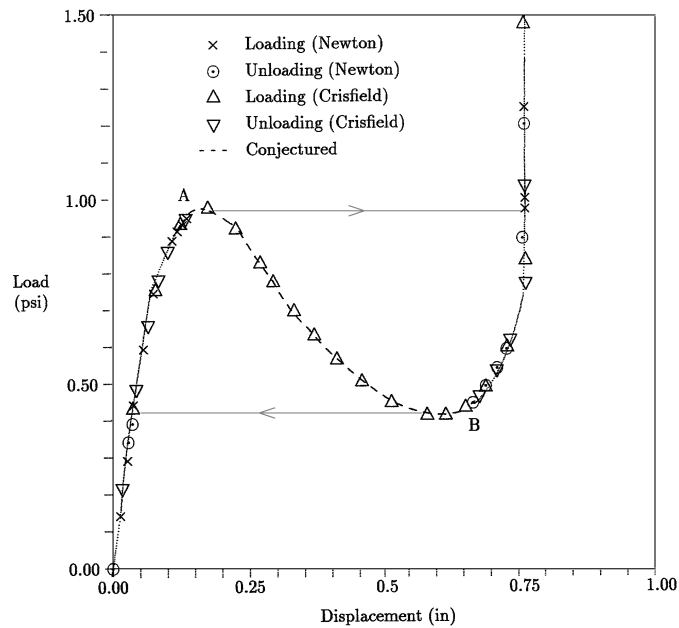


Figura 6.6: Confronto tra metodi ad incremento di carico (Newton) e di lunghezza d'arco (Crisfield) nella risoluzione del problema dell'instabilità a scatto

6.1.3 Applicazione: analisi dei longheroni

I longheroni sono componenti strutturali soggetti prevalentemente a sforzi di taglio. Nel progetto di questi elementi, si ammette normalmente che si possano caricare oltre il limite di stabilità, o in altre parole che vadano in campo *post-critico*, che per i componenti soggetti a taglio significa sviluppare il **campo di tensione diagonale** (si veda 6.1.3). In questo caso si deve necessariamente abbandonare l'ipotesi semplificativa di spostamenti piccoli e cercare la soluzione di un problema non lineare. La possibilità, in ambiente *modeFrontier*, di calcolare la superficie di risposta di un sistema fisico al variare di alcuni parametri permette di applicare un blocco aggiuntivo al ciclo iterativo di ottimizzazione illustrato nei capitoli precedenti.

In altre parole questa tecnica consente di svolgere un numero limitato di analisi non lineari mirate. Queste analisi rappresentano dei punti nell'iperspazio delle soluzioni in funzione delle variabili. Con la tecnica del modello surrogato è possibile calcolare una superficie in questo spazio i cui punti corrispondano ad una ragionevole stima delle risposte del sistema fisico alle variabili immesse. Nel caso in esame, svolgendo delle simulazioni su diversi campioni dei quali vengano variate le dimensioni principali, sottoposti a taglio puro, dei quali venga rilevato lo spostamento verticale. Una volta calcolata la curva dello spostamento massimo in funzione della tensione tangenziale applicata, da essa deriva la legge

$$G(\tau) = \frac{\gamma}{\tau} \quad .$$

Una volta costruita la superficie di risposta, questa relazione diviene disponibile per un pannello di dimensioni qualsiasi, ed in sostanza si può inserire nello schema di *mF* un blocco che, dato il livello di sollecitazione e le dimensioni del componente in esame, fornisce il valore effettivo della rigidezza a taglio. Se la differenza baia per baia tra le due rigidezze, quella calcolata con la superficie di risposta e quella usata nell'analisi, fosse superiore ad un valore di tolleranza, il ciclo riprenderebbe dalla fase pre-processo, con la creazione di un modello le cui proprietà vengano modificate sulla scorta dell'uscita del blocco non lineare. Questa strategia ha l'inconveniente di richiedere un calcolo che durerà n volte di più di un ciclo normale, se n è il numero di cicli richiesti per la convergenza del calcolo della rigidezza a taglio. Con riferimento al diagramma riportato in figura 4.2, il flusso logico di programma si modificherebbe secondo quanto indicato in figura 6.7. Il blocco **Matlab** aggiuntivo scrive un altro *file* di sessione che va sovrapporsi al precedente, generando un modello con caratteristiche modificate. Per maggior chiarezza del diagramma si sono omessi alcuni blocchi di flusso di dati, che in realtà ci sono sempre, in quanto il resto del ciclo rimane invariato.

Un altro metodo per affrontare il problema è di svolgere una *SOL106* al posto della *SOL101* nel ciclo principale, creando un materiale la cui rigidezza a taglio sia non lineare.

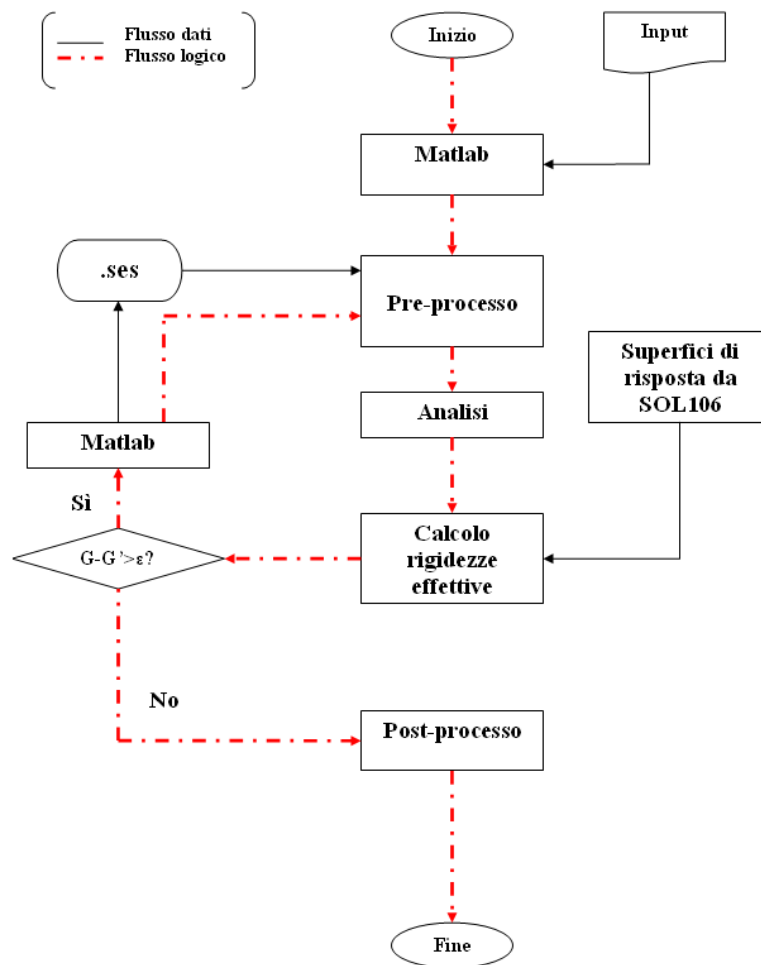


Figura 6.7: Diagramma di flusso modificato

re, corrispondente alla legge trovata con le superfici di risposta. Per fare questo occorre creare un modello di longerone formato da due elementi che lavorano in parallelo, uno di tipo *shell* che lavori a trazione e compressione, l'altro di tipo *shear* che lavori a taglio. Per modificare la rigidezza del componente si agisce sullo spessore effettivo della piastra *shear*. Questo metodo è stato tratto da (25), al quale si rimanda per approfondimenti.

Il campo di tensione diagonale

Come noto dalla letteratura (si veda il testo fondamentale del Kuhn, (16)), un pannello rettangolare, sottoposto ad una sollecitazione di taglio puro, vincolato ad aste in ogni lato (**inquadrettato**) è in grado di sopportare dei carichi molto maggiori del limite di stabilità a

taglio sviluppando un campo di tensione diagonale. Il principio del fenomeno è illustrato in figura 6.8.

Per sollecitazioni al di sotto del carico critico di instabilità, le tensioni principali sono uguali in modulo e di segno opposto ed orientate a 45° rispetto ai lati del pannello; questa è la condizione del normale campo di tensione tangenziale. Al crescere della forza tagliente, la tensione principale compressiva raggiunge il valore critico; a questo punto gli ulteriori incrementi di carico vengono compensati dalla tensione principale trattiva, chiamata **tensione diagonale**, gli assi principali non sono più orientati esattamente a 45° ed il pannello sviluppa delle ondulazioni. I pannelli reali sono soggetti al cosiddetto campo di tensione diagonale incompleto, nel quale è presente, per quanto piccola rispetto alla tensione diagonale, una tensione principale di compressione. Il caso ideale, studiato dal Wagner¹, assume che il carico critico a taglio del pannello sia nullo, e che quindi tutta la sollecitazione venga sopportata dalla tensione diagonale. Il Kuhn ha introdotto l'uso del *fattore di tensione diagonale* k come indice del tipo di sollecitazione:

$$k(\tau) = 1 - \frac{\tau_{crit}}{\tau} \quad .$$

Se il fattore è nullo il campo di tensioni è tangenziale, per fattore unitario è diagonale completo. Mediante questo, tutte le formule ricavate dal Wagner vengono corrette per adattarsi meglio ai risultati sperimentali.

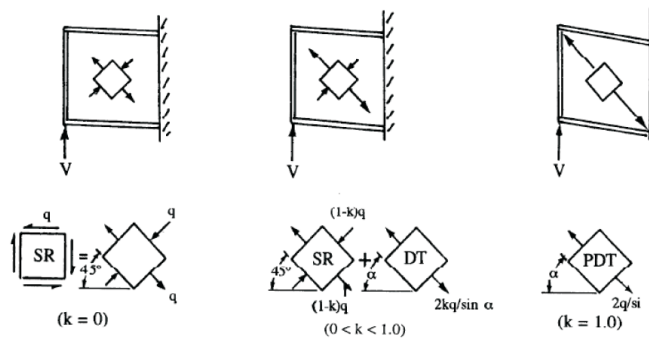


Figura 6.8: Da sinistra a destra: campo di tensione tangenziale, diagonale incompleto e puro

¹Herbert Wagner. *Flat sheet metal girders with very thin metal web*. NACA TM 604,605 e 606, anno 1931

6.2 I materiali compositi

6.2.1 Introduzione

La possibilità di sfruttare le vantaggiose caratteristiche dei materiali compositi è molto appetibile per il progettista strutturale. Introdotti relativamente da poco, i materiali compositi – in particolare le fibre di carbonio con matrice epossidica, costituente innovativo più usato nei recenti sviluppi del campo aeronautico – consentono di ottenere una struttura più leggera a parità di robustezza. Le complicazioni teoriche nello studio dei materiali di questo tipo, che differiscono dai metalli perché reagiscono in modo diverso a sollecitazioni in direzioni diverse, caratteristica nota col nome di *anisotropia*, sono compensate dai vantaggi in termini di peso della struttura. Questo ed i successivi paragrafi vogliono essere una panoramica delle problematiche derivanti dall'introduzione dei compositi nella progettazione strutturale, fermi restando gli innegabili vantaggi: miglior rapporto resistenza meccanica-peso, semplicità di lavorazione ed assemblaggio in quanto i manufatti in composito tendono all'integrazione di componenti diverse ed elevatissima resistenza a fatica trattiva. Sull'altro piatto della bilancia si deve citare la scarsa resistenza ad impatto, la sensibilità alla fatica compressiva a causa delle delaminazioni, il costo elevato e la notevole sensibilità alle condizioni climatiche, specialmente all'umidità.

6.2.2 Teoria dei laminati

Per poter studiare un materiale **ortotropo**, ovvero con tre piani di simmetria della risposta alle sollecitazioni, occorre introdurre la cosiddetta *teoria dei laminati*. Questa parte da delle ipotesi fondamentali sul comportamento del materiale:

- il laminato è composto da un certo numero di lamine impilate, collegate da un legame rigido
- il sistema si deformerà poco nella direzione trasversale ($\varepsilon_z \ll t$) e le sezioni ortogonali al piano medio resteranno piane in seguito alla deformazione
- le singole lamine costituenti, soggette a stato piano di tensione, sono isotrope nella direzione trasversale (*isotropia planare*)

Si distingue un sistema di riferimento globale (x, y, z) del laminato ed uno locale $(1, 2, 3)$ della lamina, nel quale la prima direzione sarà quella di orientamento delle fibre (si veda figura 6.9, nella quale gli assi z e 3 sono omessi).

Queste relazioni vettoriali derivano dall'assunzione

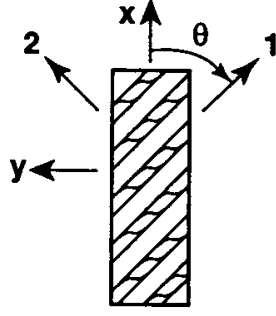


Figura 6.9: Sistemi di riferiment locale e globale del laminato

Dato un generico materiale iperelastico e perfettamente anisotropo, caratterizzato da una matrice costitutiva simmetrica, che lega le 6 componenti $\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{yz}, \gamma_{xz}, \gamma_{xy}$ della deformazione con le 6 componenti $\sigma_x, \sigma_y, \sigma_z, \tau_{yz}, \tau_{xz}, \tau_{xy}$ della tensione, composta da 21 costanti elastiche, si assume che per esso valvano le ipotesi di isotropia planare e stato piano di tensione. La prima condizione determina una riduzione da 21 a 5 delle costanti elastiche, la seconda permette di eliminare tre righe e tre colonne della matrice (si trascurano infatti $\sigma_z, \tau_{yz}, \tau_{xz}$).

Per la singola lamina, soggetta a *plane stress* ed isotropa nel piano trasversale, valgono le seguenti

$$\begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{pmatrix} = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{21}}{E_2} & 0 \\ -\frac{\nu_{12}}{E_1} & \frac{1}{E_2} & 0 \\ 0 & 0 & \frac{1}{G_{12}} \end{bmatrix} \cdot \begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{pmatrix}, \quad (6.2)$$

e, invertendo la matrice, si trova il legame fra sforzi e deformazioni

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{pmatrix} = \begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{21} & Q_{22} & 0 \\ 0 & 0 & Q_{66} \end{bmatrix} \cdot \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{pmatrix} = [Q(0)] \cdot \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{pmatrix}. \quad (6.3)$$

La matrice costitutiva ridotta è formata da cinque costanti elastiche, quattro delle quali indipendenti. Queste sono:

E_1, E_2 moduli di Young nella direzione 1 o *longitudinale* e 2 o *trasversale lunga*

ν_{12}, ν_{21} coefficienti di Poisson, il primo governa la contrazione in direzione 2 a causa di una tensione in direzione 1, viceversa il secondo

G_{12} rigidità a taglio nel piano (1,2)

La terza direzione, diretta secondo lo spessore, viene chiamata *trasversale corta*; per brevità si indicano le tre direzioni con L, LT ed ST .

Grazie alla simmetria della matrice, le prime quattro costanti sono legate dalla relazione

$$\frac{\nu_{21}}{E_2} = \frac{\nu_{12}}{E_1} \quad .$$

Le equazioni 6.3 sono valide per una lamina avente l'asse longitudinale orientato come l'asse principale x del laminato. Se invece l'angolo θ tra assi locale e globale fosse non nullo, occorrerebbe operare una trasformazione vettoriale mediante una matrice $[R(\theta)]$, la quale contiene termini in seno e coseno dell'angolo di rotazione, ottenendo una matrice Q senza termini nulli, caratterizzante quindi un comportamento della lamina non più ortotropo:

$$[Q(\theta)] = [R(\theta)] \cdot [Q(0)] \cdot [R(\theta)]^T = \begin{bmatrix} \bar{Q}_{11} & \bar{Q}_{12} & \bar{Q}_{16} \\ \bar{Q}_{21} & \bar{Q}_{22} & \bar{Q}_{26} \\ \bar{Q}_{61} & \bar{Q}_{62} & \bar{Q}_{66} \end{bmatrix} \quad (6.4)$$

Il legame sforzo-deformazione di una generica lamina orientata di un angolo θ rispetto agli assi principali si scrive dunque

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{pmatrix} = [Q(\theta)] \cdot \begin{pmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{pmatrix} \quad . \quad (6.5)$$

Per passare dalla relazione 6.5 a quella per il laminato completo, bisogna immaginarlo come una serie di lamine che lavorano in parallelo, ciascuna delle quali reagisce alla deformazione del laminato con un vettore di tensioni $\sigma_{x,k}, \sigma_{y,k}, \tau_{xy,k}$, dove k è l'indice della lamina.

Per le ipotesi fatte, tutte le lamine hanno una deformazione proporzionale alla loro distanza z dal piano medio:

$$\varepsilon_i(z) = \varepsilon_{i0} + z \cdot \chi_i \quad .$$

Nella relazione appena scritta compare la deformazione del piano medio ε_{i0} e la curvatura

del laminato χ_i . In base a quanto detto

$$\begin{pmatrix} \sigma_{x,k} \\ \sigma_{y,k} \\ \tau_{xy,k} \end{pmatrix} = [Q(\theta_k)] \cdot \begin{pmatrix} \varepsilon_{x,0} + z_k \cdot \chi_x \\ \varepsilon_{y,0} + z_k \cdot \chi_y \\ \gamma_{xy,0} + z_k \cdot \chi_{xy} \end{pmatrix} . \quad (6.6)$$

Riferendosi agli indici di carico del laminato, definiti da equazioni del tipo

$$\begin{aligned} N_x &= \int_{\frac{t}{2}}^{-\frac{t}{2}} \sigma_x dz = \sum_{k=1}^n \int_{z_{k-1}}^{z_k} \sigma_{x,k} dz \\ M_x &= \int_{\frac{t}{2}}^{-\frac{t}{2}} z \sigma_x dz = \sum_{k=1}^n \int_{z_{k-1}}^{z_k} z \sigma_{x,k} dz , \end{aligned}$$

è possibile ricondursi alle seguenti relazioni generali:

$$\begin{pmatrix} N_x \\ N_y \\ N_{xy} \end{pmatrix} = [A] \cdot \begin{pmatrix} \varepsilon_{x,0} \\ \varepsilon_{y,0} \\ \gamma_{xy,0} \end{pmatrix} + [B] \cdot \begin{pmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{pmatrix} , \quad (6.7)$$

$$\begin{pmatrix} M_x \\ M_y \\ M_{xy} \end{pmatrix} = [B] \cdot \begin{pmatrix} \varepsilon_{x,0} \\ \varepsilon_{y,0} \\ \gamma_{xy,0} \end{pmatrix} + [D] \cdot \begin{pmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{pmatrix} , \quad (6.8)$$

nelle quali i termini delle matrici $[A]$, $[B]$, $[D]$, derivanti dagli integrali delle tensioni, hanno le seguenti espressioni:

$$A_{ij} = \sum_k Q_{ij}(\theta_k) \cdot (z_k - z_{k-1}) \quad (6.9)$$

$$B_{ij} = \frac{1}{2} \cdot \sum_k Q_{ij}(\theta_k) \cdot (z_k^2 - z_{k-1}^2) \quad (6.10)$$

$$D_{ij} = \frac{1}{3} \cdot \sum_k Q_{ij}(\theta_k) \cdot (z_k^3 - z_{k-1}^3) \quad (6.11)$$

La matrice $[A]$ del laminato è formalmente identica alla matrice $[Q]$ della lamina nell'equazione 6.3, in particolare il laminato esibirà comportamento ortotropo se si annulleranno i termini di accoppiamento del taglio con la deformazione ad estensione A_{16} e A_{26} . Questo si può ottenere realizzando un laminato **bilanciato**, nel quale cioè tutte le lamine compaiono a coppie di angoli uguali e contrari; in questo modo, nella sommatoria 6.9, per i termini di accoppiamento taglio-estensione si ha una serie di contributi uguali e contrari che danno contributo complessivamente nullo. Un laminato che presenta invece questo accoppiamento sviluppa tensioni di taglio anche se soggetto a trazione pura, come si vede in figura 6.10(a).

La matrice $[B]$ è detta di accoppiamento tra estensione e flessione e determina un effetto riassunto in figura 6.10(b). Un laminato **simmetrico** ha nulli tutti i termini di questa matrice, perchè nell'equazione 6.10 scritta per questo apparirebbero soltanto addendi uguali e contrari.

È pratica comune realizzare la maggior parte dei laminati senza questo accoppiamento; ciò li rende più semplici da studiare, in quanto le equazioni 6.7 e 6.8 possono essere risolte separatamente, e previene l'inevitabile inflessione che si verificherebbe in un laminato non simmetrico durante il raffreddamento in seguito alla formatura.

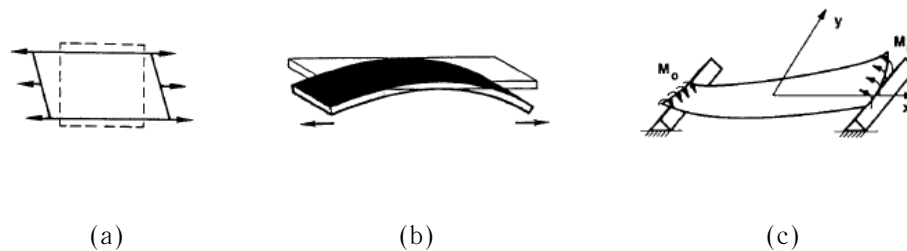


Figura 6.10: Accoppiamenti meccanici nel laminato: taglio-estensione (a); flessione-estensione (b); flessione-torsione (c)

La matrice $[D]$ lega la deformazione fuori dal piano (curvature χ_x , χ_y e χ_{xy}) del laminato con i momenti flettenti M_x , M_y e torcente M_{xy} . In particolare i termini D_{16} e D_{26} legano insieme torsione e flessione e rendono conto di un effetto visualizzato in figura 6.10(c); i laminati simmetrici, oltre a non presentare accoppiamento tra flessione e trazione, hanno anche il vantaggio di non avere accoppiamento tra torsione e flessione.

6.2.3 Alcune indicazioni sull'ottimizzazione di strutture in composito al *modeFrontier*

Nel paragrafo 6.2.2 sono stati illustrati i principali strumenti per lo studio di un materiale multistrato. In questo si analizza quali complicazioni comporterebbe applicare la procedura sviluppata nei capitoli precedenti per il progetto di un cassone alare non più in lega metallica bensì in fibra di carbonio.

Da quanto si è detto finora, appare evidente che i materiali compositi consentano di migliorare l'efficienza della struttura al prezzo di una complicazione teorica. Il vantaggio, al fine di modificare la rigidità, di poter variare non soltanto lo spessore ma anche l'orientamento delle lamine costituenti introduce nuove variabili al problema.

È pratica comune realizzare pannelli in composito con lamine tutte dello stesso spessore e disposte solamente ad angoli di 0° , 90° , $+45^\circ$ e -45° . Inoltre, per realizzare un laminato

simmetrico e bilanciato, in modo da eliminare tutti gli accoppiamenti visti in precedenza, le lamine a $+45^\circ$ e -45° devono essere in egual numero. Dunque per ogni elemento della struttura che abbia uno spessore distinto vengono introdotte *due* variabili in più: a parte il numero totale di lamine (che sostituisce la dimensione trasversale dell'elemento), occorre conoscere la percentuale di lamine a 0° ed a $\pm 45^\circ$, essendo determinata di conseguenza la percentuale di quelle a 90° .

Nel modello adottato per il ciclo iterativo descritto nel capitolo 4 si usano sei spessori: dei due pannelli e dei rispettivi correnti e dei due longheroni. Dunque per adottare un costituente composito verrebbero introdotte 12 nuove variabili, a meno che non si scelga di operare delle semplificazioni come per esempio fissare le percentuali di lamine a 0° e/o a $\pm 45^\circ$. In questo modo si sceglie una configurazione che vada mediamente bene per il componente e si ottimizza soltanto il numero di lamine.

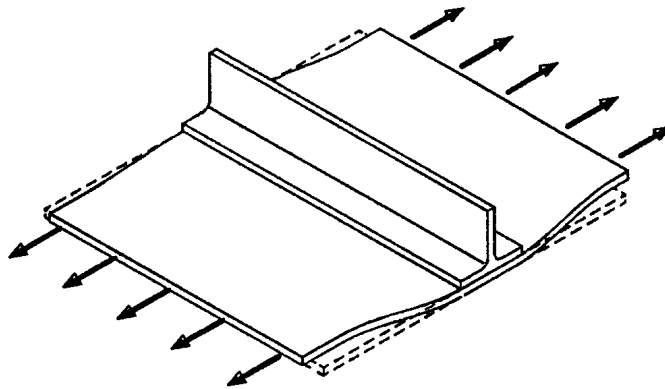


Figura 6.11: Sforzi interlaminari causati da una eccessiva differenza di coefficienti di Poisson di due laminati

Un'altro fattore da tenere in conto è la sequenza di impilamento o *stacking sequence*, che deve rispettare certi requisiti tecnologici, suggeriti in (20):

- realizzare laminati simmetrici e bilanciati, ove non vi siano esigenze diverse
- usare almeno un gruppo $45/90/-45$ sulla superficie esterna, per migliorare la resistenza ad impatto
- non usare più di quattro lamine con la stessa orientazione per non indebolire eccessivamente i bordi, più di una se orientata a 90°

- non impilare due lamine adiacenti orientate reciprocamente ad un angolo maggiore di 60° , perchè aumenterebbe la probabilità di avere fratture microscopiche a causa delle tensioni residue durante la fase di cura e raffreddamento
- non realizzare laminati con meno del 10% di lamine a 0° , 90° o $\pm 45^\circ$ e con meno di sette lamine
- evitare una eccessiva differenza tra i coefficienti di Poisson di due laminati diversi incollati insieme che potrebbe causare *stress* interlaminari (si veda figura 6.11)

Questi requisiti possono essere posti sottoforma di vincoli, oppure si può introdurre un *indice tecnologico* da minimizzare, prevedendo di aumentarlo consistentemente nel caso venisse violata una delle condizioni illustrate.

Il lavoro presente di tesi nasce dal desiderio di sviluppare uno strumento di semplice uso per il progetto strutturale preliminare di un cassone alare: le caratteristiche fondamentali di tale strumento devono essere la semplicità e l'immediatezza d'uso, la possibilità di analizzare rapidamente più configurazioni, la capacità di ottimizzare la struttura con requisiti contrastanti e vincoli di natura tecnica, l'uso di strumenti di analisi standard.

La sessione di analisi ha come obiettivo l'individuazione dei flussi di carico interni alla struttura alare, necessario punto di partenza per studi di livello successivo, come ad esempio analisi di dettaglio di componenti (centine, longheroni).

Con questo intento e quello di sviluppare una geometria comunque rappresentativa della realtà, si è scelto di realizzare un modello *FEM* tridimensionale caratterizzato da una griglia piuttosto rada (*coarse mesh*), fatta di elementi di tipo guscio e altri di tipo trave. Questa fase è stata illustrata nel capitolo 2.

Una volta realizzato il processo di creazione della struttura, è stato necessario valutare un insieme di condizioni di carico che rappresentasse in prima battuta le sollecitazioni più importanti per il dimensionamento. Questa selezione, basata sui regolamenti aeronautici, è stata affrontata nel capitolo 3.

I modelli realizzati e il sistema concepito per generarli ed analizzarli, come estesamente illustrato nel capitolo 4, forniscono dei risultati che permettono di valutare correttamente gli stati di tensione e deformazione ai quali lavora la struttura. È stata inoltre effettuata la verifica funzionale del processo ed una valutazione delle prestazioni in termini di tempi di calcolo: nelle tabelle 5.2, 5.4 e 5.5 si vede che sono indicati tempi medi per il completamento inferiori al minuto per configurazione.

Intuendo la portata del lavoro che si profilava, si è cercato di realizzare un'architettura *modulare*, formata da un insieme di funzionalità facilmente sostituibili e migliorabili in un

secondo momento da chi scrive o da altri, consci del fatto che l'uso di applicazioni di questo tipo da parte di chi non le ha realizzate direttamente non è mai immediato e con l'intenzione di rendere la lettura e la modifica dei codici messi a punto operazioni quanto più semplici possibile.

Una volta completato il processo di calcolo sul problema di volta in volta richiesto, che può essere uno studio di sensibilità o la valutazione dell'architettura preferibile dati i requisiti, si possiede, per ogni modello tridimensionale generato, un insieme di informazioni che partono da tutti i *file* di ingresso ed uscita dell'analisi degli elementi finiti per arrivare a diagrammi sull'andamento di sollecitazioni e margini di sicurezza in apertura, oltre che una tabella nell'ambiente dell'ottimizzatore contenente tutti i valori numerici relativi ad ogni *design* valutato, riorganizzabili in grafici e diagrammi che forniscono ulteriori informazioni sugli andamenti delle variabili e degli obiettivi, come è spiegato negli esempi presentati nel capitolo 5. Un volume di dati così vasto ha comportato scelte critiche per quanto riguarda i *file* da tenere in memoria per ogni soluzione elaborata. In particolare si è scelto di cancellare quelli che contengono il modello tridimensionale perché molto ingombranti; questo comporta l'inconveniente, nel caso si desiderasse visualizzare col *post processor* le tensioni o le deformazioni provocate sul cassone da una qualsiasi condizione di carico, di dover eseguire nuovamente la sessione che crea il modello, il che rende non immediata la visualizzazione e può rallentare il confronto fra più soluzioni. D'altro canto questo riduce lo spazio occupato in memoria da ogni configurazione a *un decimo* di quello che occuperebbe altrimenti ed è facile intuire che, avendo a che fare con processi che coinvolgono migliaia di *design*, la necessità di ridurre le richieste di memoria è di vitale importanza.

Qualora la presente tesi avesse un seguito, sarà possibile operare sul singolo elemento costituente l'insieme, sostituendolo con qualcosa di più accurato, in modo da migliorarne l'affidabilità pur rimanendo nell'ambito dello schema consolidato. Ad esempio, in futuro si potranno sostituire alcune relazioni con delle formule più precise o con dati provenienti da una *response surface*, oppure cambiare le caratteristiche meccaniche del materiale o il tipo di analisi svolta dal solutore ad elementi finiti, come si è discusso nel capitolo 6, o realizzare un'interfaccia grafica amichevole in modo da rendere più semplice l'uso del codice a chi non avesse grande dimestichezza con le procedure *software*.

APPENDICE A

BREVE GUIDA ALL' USO DEL SOFTWARE

Il software sviluppato nel presente lavoro comprende una parte, svolta in ambiente Matlab, che crea dei *file* di sessione scritti nel linguaggio di programmazione di Patran, il *Patran command language* (PCL), ed un cosiddetto progetto di *modeFrontier* che gestisce il ciclo di ottimizzazione.

A.1 Principali estensioni dei file

- .bdf** file di input di **Nastran**, creato da **Patran**
- .db** file dati di Patran. Contiene tutte le informazioni sulla geometria del modello 3D
- .exe** eseguibile di un qualsiasi programma
- .f06** output testuale di Nastran, vi sono riportati i valori numerici dei risultati
- .m** file Matlab
- .prj** file modeFrontier
- .rpt** file di testo prodotto da Patran su richiesta dell'operatore, contiene una serie di informazioni riguardanti il modello
- .ses** file di input di Patran, contiene una sequenza di comandi che il programma eseguirà
- .xdb** file di output di Nastran, che permette di sfruttare il postprocessore per visualizzare i risultati

A.2 Il file .prj di modeFrontier

Il *modeFrontier* (da ora in avanti *mF*) è un ottimizzatore multiobiettivo generico, in altre parole occorre che l'utente definisca lo spazio nel quale agirà il *software* (ad esempio un sistema fisico con le proprie leggi), e questo ottimizzerà le variabili numeriche in gioco, fornite coerentemente con lo spazio e le leggi specifiche.

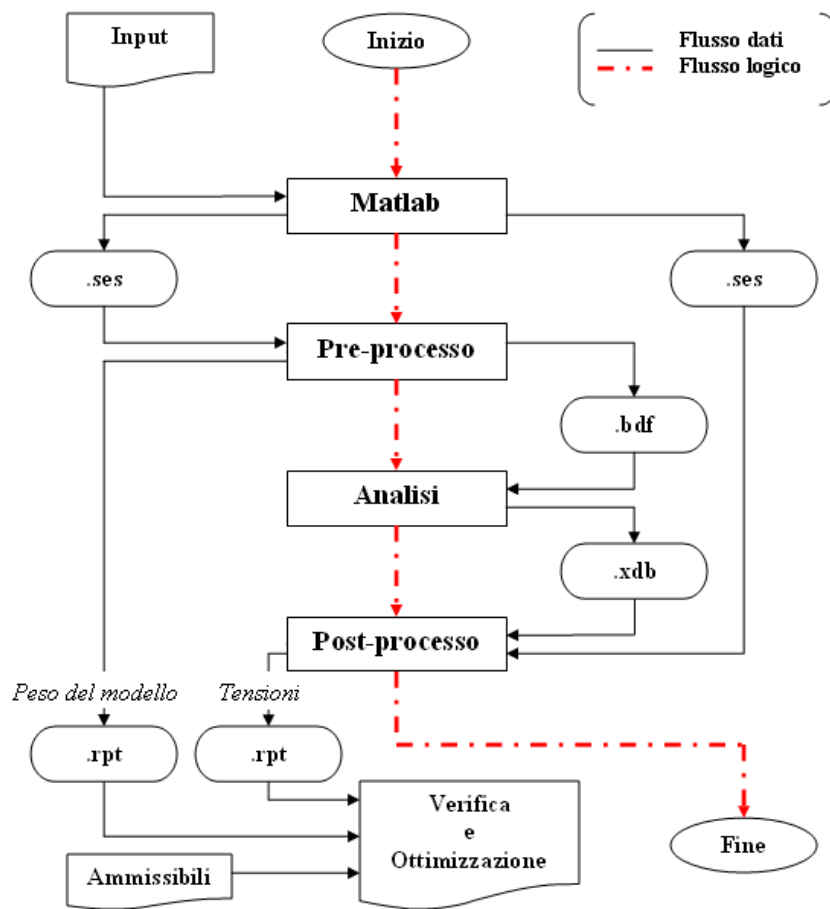
Per fare ciò il programma è in grado, con le modalità che si vedranno più avanti, di interfacciarsi ad altri *software* e di applicare su scelta dall'utente un determinato algoritmo di ottimizzazione, tra i quali spiccano gli algoritmi genetici.

L'interfaccia grafica del *mF* è un diagramma a blocchi e connettori formato da nodi di esecuzione di programmi ausiliari ed altri di acquisizione dati ('lettura' di valori numerici da un *file* testuale). Il *mF* fornisce dei nodi dedicati ad un *software* specifico (ad esempio esistono i nodi **Matlab**, **Excel**, **Abaqus**, solo per citarne alcuni), ma è altresì possibile richiamare un qualsiasi programma tramite uno *script* in DOS. Nello schema che costituisce il *file* .prj si distingue il **flusso logico**, cioè la sequenza delle operazioni da realizzare, ed il **flusso di dati**, costituito da variabili interne e *file* esterni in ingresso e in uscita dai nodi.

Dunque il *mF* effettua una sequenza logica di operazioni per un certo numero di volte, servendosi di variabili numeriche interne al *mF* stesso e di *file* utili per il funzionamento dei programmi richiamati all'interno del progetto. Per gestire questo volume di dati l'ottimizzatore crea una cartella nella quale ne alloca altre per contenere gli ingressi e le uscite. Ne vengono create tante quanti sono i nodi nel progetto ed in ognuna di queste vengono riversati i *file* ed un resoconto dei dati numerici in ingresso ed in uscita dal nodo. In questo modo è possibile andare a vedere, dopo l'esecuzione del progetto, il flusso in ogni nodo e capire, in caso di fallimento, quale nodo ha creato problemi ed in che modo.

Al momento di eseguire l'operazione dettata dal nodo, viene creata la cartella di pertinenza e vi vengono riversati i dati ed i *file* necessari, dopodiché viene lanciato il programma relativo al nodo in questione; per far sì che l'uscita di un certo nodo si vada a trovare nella cartella del nodo successivo occorre dire al *mF* che deve copiare detta uscita dalla cartella del processo precedente alla cartella del processo attuale, mediante il nodo *transfer file*. In questo modo per esempio si trasferisce il *file* di sessione dalla cartella 'Matlab' alla cartella 'Patran', ed il file .bdf da quest'ultima alla cartella 'Nastran'.

Per estrarre valori numerici di interesse per il calcolo dai *file* di testo il *software* va a cercare in una certa posizione all'interno dello stesso il valore numerico che gli serve. Questa posizione, definita dall'operatore, potrebbe essere assoluta, per esempio 'alla terza riga e seconda colonna', oppure relativa, come 'alla terza riga e seconda colonna dopo una certa parola chiave'. Con questo sistema è possibile leggere i valori dell'analisi Nastran direttamente dal .f06 o dal .rpt .

Figura A.1: Diagramma di flusso del progetto *modeFrontier*

In figura A.1 è riportato il diagramma semplificato del progetto di ottimizzazione del cassone alare. Nel blocco 'input' vengono generati i valori delle variabili che verranno riversate nel *workspace* di Matlab, nel cui nodo si generano due *file* di sessione forniti al primo nodo Patran ('preprocesso': generazione del modello e creazione del file .bdf) ed al secondo ('post-processo': lettura dei risultati dell'analisi), si veda per maggiori dettagli il paragrafo A.3. Il nodo Nastran ('analisi') riceve il .bdf relativo al modello costruito con le variabili del blocco 'input' ed esegue l'analisi agli elementi finiti. In 'Verifica e ottimizzazione' vengono analizzati gli .rpt del modello ed imposti gli obiettivi ed i vincoli; nel caso in esame l'obiettivo è minimizzare il peso del modello col vincolo che tutte le tensioni stiano al di sotto dei valori ammissibili.

Lo schema riproduce nel complesso le operazioni che l'utente eseguirebbe manualmente se dovesse disegnare con Patran un certo modello con caratteristiche specifiche, svolger-

ne l'analisi strutturale e verificare che le tensioni e le deformazioni in esso siano accettabili. Il ciclo di ottimizzazione si realizza mediante la ripetizione di questa sequenza tante volte quanto richiesto, variando i valori geometrici di partenza in base ai criteri propri degli algoritmi genetici: a partire da un gruppo di configurazioni o individui 'capostipiti', ne viene valutata la bontà mediante una funzione di *fitness* (forma); a questo punto si crea una nuova generazione facendo in modo che gli individui più in forma abbiano una maggiore probabilità di figliare, così da creare una generazione di migliore qualità rispetto alla precedente. I nuovi individui nascono per mutazione, incrocio o duplicazione delle configurazioni migliori della generazione precedente, mentre a quelli che non rispettano i vincoli richiesti (*unfeasible* ovvero inaffidabili) viene introdotta una penalità nel calcolo della forma, riducendone di fatto la possibilità di riprodursi. È possibile trovare una spiegazione dettagliata e chiara degli algoritmi genetici in (11), testo al quale rimandiamo per approfondimenti.

Generazione dopo generazione, ci si avvicina alla *frontiera di Pareto*, luogo delle configurazioni di massimo assoluto o dominanti; il processo si arresta quando arriva ad una certa generazione definita dall'utente.

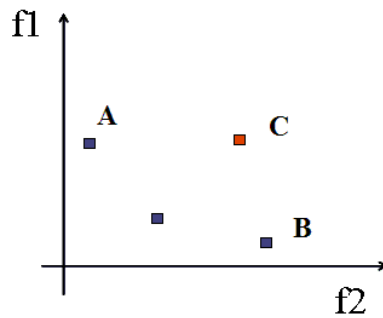


Figura A.2: La frontiera di Pareto

La frontiera è il luogo dei punti soluzione tali per cui nessun componente può essere migliorato in un obiettivo senza peggiorarlo in un altro. In figura A.2 si vede, in un problema con due funzioni f_1 ed f_2 da minimizzare¹, che la configurazione **C** è dominata, mentre le configurazioni in blu giacciono sulla frontiera.

Per un problema del genere il punto **A** domina sul punto **C** se sono vere le seguenti relazioni:

$$\begin{cases} f_1(A) \leq f_1(C) \text{ e } f_2(A) \leq f_2(C) \\ f_1(A) < f_1(C) \text{ o } f_2(A) < f_2(C) \end{cases}$$

¹Per esempio potrebbe trattarsi di una trave caricata della quale si voglia minimizzare il peso e la freccia massima

I punti sulla frontiera sono di ottimo non dominati, cioè non esistono altri punti che dominano sui punti della frontiera. Se questo accadesse, il punto che domina su uno della frontiera lo sostituirebbe sulla frontiera stessa.

A.3 Il codice Matlab

È formato da quattro gruppi distinti:

AVVIO.m richiamato da *mF*, contiene i dati geometrici che non verranno modificati (la forma in pianta dell'ala), i percorsi verso i *file* che contengono le coordinate adimensionali dei profili alari e di quelli (sessione e database di Patran) che verranno creati; contiene inoltre il richiamo agli altri due blocchi di codice.

PROGRAMMA.m crea il primo *file* di sessione. Questo contiene i comandi per generare la geometria di partenza del cassone e la *mesh*, assegnare le proprietà ai pannelli ed alle travi, creare gli MPC sui quali verranno applicati i carichi, generare un .rpt col peso del modello ed un .bdf da fornire al Nastran. Man mano che la scrittura del .ses procede, il codice esegue diverse operazioni subordinate, quali il calcolo delle caratteristiche della sollecitazione relative ad una certa condizione di volo, la determinazione delle proprietà (spessori, aree, momenti di inerzia ed *offset*) in una baia qualsiasi conoscendo queste alla radice, al *kink* ed all'estremità e la legge di rastremazione (supposta lineare) e la creazione di una matrice di numeri di identificazione che permette di localizzare ogni elemento nella struttura (questa funzione è fondamentale per richiedere gli sforzi agenti nelle zone di controllo della struttura). Si rimanda all'appendice B per approfondimenti.

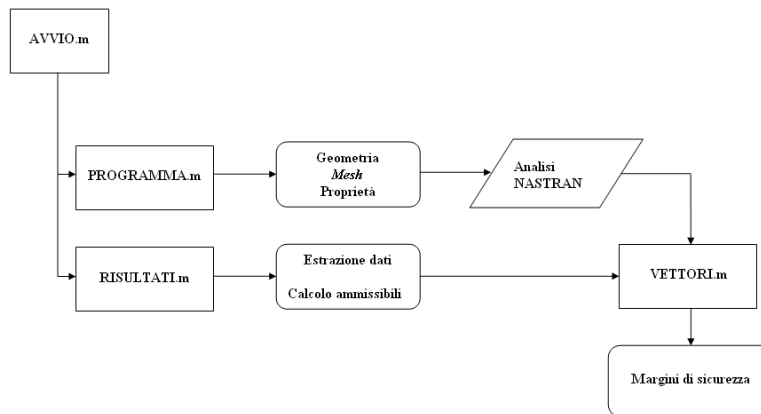
RISULTATI.m questa parte di codice scrive il secondo .ses, che comanda a Patran di riferire l'uscita di NASTRAN al modello e di creare una serie di .rpt contenti le tensioni negli elementi di controllo. Inoltre calcola gli sforzi ammissibili di ogni gruppo soluzione per ogni tipo di sollecitazione considerata. Si veda l'appendice C per maggiori dettagli.

VETTORI.m confronta gli ammissibili e le sollecitazioni baia per baia e crea dei vettori di margini di sicurezza in apertura, come spiegato estesamente in appendice C

Questi blocchi creano due *file* sessione che verranno forniti, mediante il modeFortier, in ingresso al precompilatore. **PROGRAMMA.m** crea inoltre via Patran un file .bdf per l'analisi Nastran vera e propria. Uno schema logico del processo è rappresentato in figura A.3.

Per creare i .ses si sfruttano i comandi del Matlab **fopen** e **fclose** per aprire e chiudere un generico *file* e **fprintf** per scrivere nello stesso una riga di testo con la sintassi del *PCL*. Quest'ultimo comando dà anche la possibilità di inserire nella riga scritta il valore di una qualsiasi variabile numerica o stringa di caratteri; sarà pertanto possibile riempire la struttura di base con i valori geometrici voluti di volta in volta.

A titolo di esempio si riportano le righe di comando per scrivere un *file* di testo (.txt) contenente una frase con una certa variabile numerica:

Figura A.3: Schema logico dei blocchi **Matlab**

```

fid=fopen('C:/messaggio.txt','w');
fprintf(fid,'Il valore di pi greco fino al decimo numero dopo la virgola è %1.10f',pi);
fclose(fid);

```

Quanto scritto si può riassumere in questo modo: apri `messaggio.txt` con l'intento di scriverci qualcosa (`'w'`), la riga da comporre è la stringa riportata, nella quale bisogna inserire il valore numerico di π in formato col punto (*fixed-point*) con una cifra prima e dieci dopo la virgola, poi chiudere il *file*.

A.4 Il Patran command language

Il Patran è un *pre/postprocessore* del Nastran. Questo vuol dire che è in grado di creare un modello tridimensionale per l'analisi, svolta comunque da Nastran, mediante l'interfaccia grafica ed è altresì capace dopo l'analisi di realizzare dei resoconti visivi dei risultati (rappresentazione degli sforzi e degli spostamenti con l'uso di colori, vettori, eccetera).

Ogni volta che Patran viene aperto e si svolgono le sue proprie funzionalità (creazione di entità geometriche, elementi finiti, vincoli e quant'altro), viene salvata una sorta di registro contenente tutte le operazioni effettuate, nella sequenza in cui sono state svolte dall'utente, dimodoché sia possibile ricostruire da capo lo stesso modello tridimensionale senza dover ripetere manualmente gli stessi comandi. Questo registro è il già menzionato *file di sessione* del Patran.

Il codice Matlab illustrato nel paragrafo A.3 genera nel linguaggio del Patran una sessione fittizia che questo richiamerà per ricostruire la geometria e la *mesh* voluta.

Per scrivere questo codice è stato necessario farsi un'idea di massima della sintassi del PCL. Lo scopo di questo breve resoconto non è certamente spiegare comando per comando il linguaggio (per questo si rimanda al manuale (1)), bensì creare una base razionale grazie alla quale comprendere meglio il lavoro svolto, nell'ottica di un utilizzo più consapevole ed anche di un ampliamento delle funzionalità.

Normalmente, una manipolazione della geometria o della *mesh* viene richiesta mediante due righe, la prima che inizia con **STRING** e prosegue con un certo comando; nella seconda si scrive il comando per esteso, con i valori numerici pertinenti. Attraverso **STRING** si comunica al Patran che deve svolgere un'operazione diversa da quella precedente, dunque se occorresse realizzare due azioni formalmente identiche (ad esempio la creazione di due punti) non sarebbe necessario introdurre la seconda con il comando **STRING**.

Ad esempio, le righe

```
STRING sgm_surface_break_p_created_ids[VIRTUAL]
sgm_edit_surface_break_plane( "9", "Surface 5", "plane 50", FALSE, sgm_surface_break_p_created_ids
sgm_edit_surface_break_plane( "#", "Surface 9", "plane 127", TRUE, sgm_surface_break_p_created_ids
```

si leggono : prepararsi ad eseguire la divisione di superfici secondo piani di sezione, nella prima riga; dividere la superficie 5 senza cancellarla (è il significato di **FALSE**) secondo il piano 50 e numerare le entità geometriche risultanti da questa operazione a partire da 9 (dunque verranno create almeno due superfici, numerate con 9 e 10), nella seconda riga; dividere la superficie 9 secondo il piano 127 e cominciare la numerazione dal primo *ID* disponibile (richiamato col simbolo **#**) per quel tipo di entità dopodiché cancellare la superficie iniziale (**TRUE**), nella terza riga. Per alcune azioni, ad esempio per la creazione di un *mesh seed* o di un gruppo e l'assegnazione di proprietà alla *mesh*, non si deve usare il comando **STRING**.

Si noti come i parametri dell'operazione vengano racchiusi da parentesi tonde e come dentro le parentesi vi sia una sequenza di valori separati da virgole, alcuni tra virgolette, altri no. Di solito le virgolette racchiudono il numero identificativo di un'entità geometrica (preceduto da *point*, *plane*, *curve*, *surface*, *coord* per specificarne il tipo) o della *mesh* (*node* oppure *elm*), ma non è una regola, mentre i parametri di controllo del comando (ad esempio la tolleranza di un'intersezione o il numero di volte che si deve ripetere la medesima azione) sono normalmente senza virgolette.

Le operazioni sulla geometria hanno tutte quante la stessa struttura, in particolare la parte tra parentesi inizia con il numero dal quale inizierà la numerazione degli elementi creati e continua con dei valori tra virgolette che identificano le entità geometriche coinvolte nella manipolazione. Ovviamente quel che compare tra parentesi dipende dal tipo di comando. Nelle righe seguenti

```
STRING asm_delete_point_deleted_ids[VIRTUAL]
asm_delete_point( "Point 1:6 9:13:2", asm_delete_point_deleted_ids )
```

si richiede di cancellare i punti dal numero 1 al numero 6 ed i punti dispari dal 9 al 13; naturalmente non c'è nessun numero tra virgolette identificativo delle entità perché con questo comando non ne verrà creata alcuna. Si noti inoltre l'uso del simbolo ':' ad indicare tutti i numeri compresi tra i due che separa se compare una volta oppure tutti i numeri a partire dal primo fino all'ultimo incrementando di volta in volta del terzo numero se compare due volte; il comando precedente si sarebbe potuto scrivere anche in questo modo

```
STRING asm_delete_point_deleted_ids[VIRTUAL]
asm_delete_point( "Point 1 2 3 4 5 6 9 11 13", asm_delete_point_deleted_ids )
```

Un altro esempio, più articolato, è il seguente:

```
STRING sgm_transform_point_created_ids[VIRTUAL]
asm_transform_grid_translate( "10", "<-125.000000 0 0>", "Coord 0" , 9, FALSE, TRUE, @
"Point 2", sgm_transform_point_created_ids )
```

che significa: traslare il punto 2 per nove volte dopodiché cancellarlo (è il significato di **TRUE**); il primo punto creato in questo modo sarà il punto 10 (gli altri a seguire), il passo è di 125 nella direzione delle x negative del sistema di riferimento 0. **FALSE** indica che la traslazione non avverrà secondo coordinate curvilinee bensì cartesiane.

Il simbolo @, come si vede nelle righe appena scritte, serve per mandare a capo un comando particolarmente lungo. Infatti Patran non eseguirà le righe del *file* di sessione composte da più di un certo numero di caratteri. Quando viene incontrato questo simbolo la lettura del comando prosegue sino alla fine della riga successiva, e può anche essere usato in più righe consecutive per sequenze particolarmente lunghe, come accade nell'esempio a seguire:

```
INTEGER fem_create_mesh_surfa_num_nodes
INTEGER fem_create_mesh_surfa_num_elems
```

```

STRING fem_create_mesh_s_nodes_created[VIRTUAL]
STRING fem_create_mesh_s_elems_created[VIRTUAL]
fem_create_mesh_surf_4( "IsoMesh", 49152, "Surface 246", 1, @
["196.6"], "Quad4", "30000", "40000", "Coord 0", "Coord 0", @
fem_create_mesh_surfa_num_nodes, fem_create_mesh_surfa_num_elems, @
fem_create_mesh_s_nodes_created, fem_create_mesh_s_elems_created )

```

Si tratta di un comando per eseguire una *mesh* (la creazione di un singolo elemento o di un singolo nodo viene introdotto da **STRING**, una *mesh* completa da due comandi **INTEGER** e due **STRING**) isoparametrica sulla superficie 246 fatta di elementi *QUAD4*, con la numerazione dei nodi che parta da 30000 e quella degli elementi da 40000.

Il valore tra parentesi quadre e virgolette è il *global edge length* (d'ora in avanti *GEL*), fondamentale per il controllo del processo di creazione della *mesh*.

Infatti il Patran, quando gli viene richiesto di creare la *mesh*, cerca di usare, se non ha ulteriori indicazioni, elementi il più possibile regolari con lato di lunghezza pari al GEL. Se però la superficie avesse un *mesh seed* su uno o più lati, il precompilatore creerebbe gli elementi rispettando il *seed* ed imponendo sul lato **più lungo tra i rimanenti** il valore del GEL. In figura A.4 viene illustrato questo concetto.

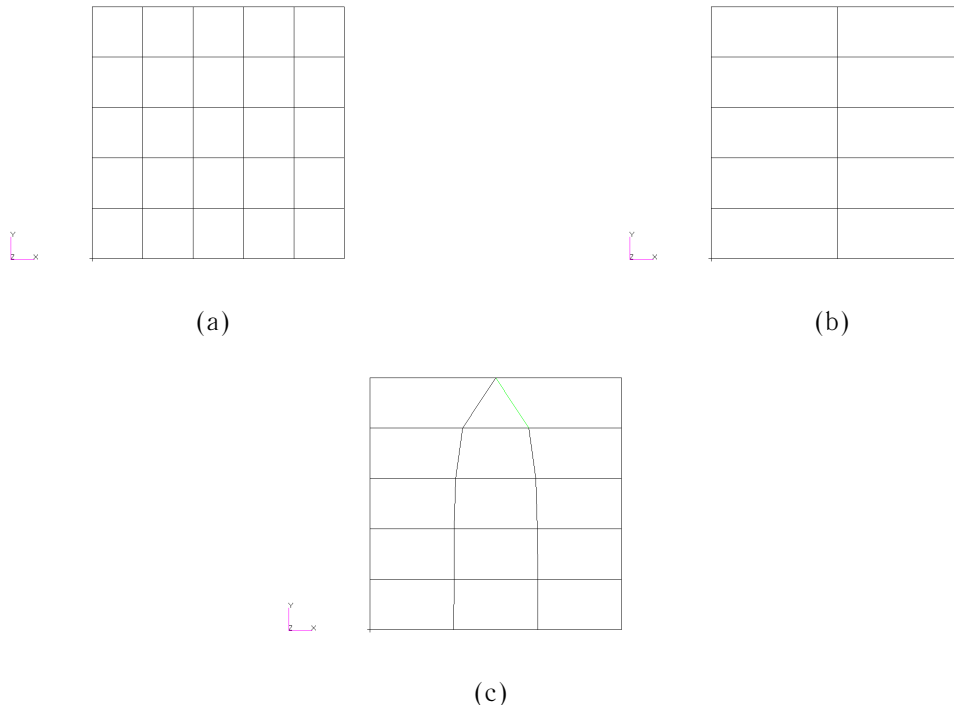


Figura A.4: Esempi di *mesh* di un quadrato 10x10: a) GEL=2 ; b) GEL=2, *mesh seed* sul lato superiore; c) GEL=2, *mesh seed* sui lati superiore ed inferiore

Come si è già detto, le azioni che non richiedano una manipolazione alla geometria o alla *mesh* non vengono introdotte da **STRING**, ad esempio la creazione di un gruppo oppure la definizione di un *seed*, oppure la seguente:

```
elementprops_create( "Centina", 51, 25, 35, 1, 1, 20, @
[13, 20, 36, 4037, 4111, 4118, 4119, 8111], [5, 9, 1, 1, 1, 1, 1, 4],@
["m:Lega", "", "8.0", "", "", "", "", ""], "Surface 1001 1003 1004" )
```

che attribuisce agli elementi *shell* relativi alle superfici 1001, 1003 e 1004 la proprietà chiamata 'Centina', che consta di uno spessore pari a 8 e di un materiale precedentemente definito col nome 'Lega'. I numeri racchiusi da parentesi quadre sono parametri che cambiano al cambiare del tipo di proprietà. Di particolare interesse in un comando di assegnazione di questo tipo è l'ultima parentesi quadra, che contiene i valori di *input* appropriati: spessore e materiale, ma anche *offset*, orientamento e caratteristiche della sezione nel caso di elementi *bar*.

APPENDICE B

IL PROCESSO DI CREAZIONE DEL MODELLO

In questa appendice si vuole analizzare nel dettaglio la procedura messa a punto dal codice **PROGRAMMA.m**, che fornisce in uscita un *file* di sessione **Patran** con le operazioni da eseguire per ottenere un modello completo, pronto per l'analisi strutturale. Predisporre un procedimento automatico per creare una geometria, generare una *mesh*, vincolare, caricare ed assegnare le proprietà è molto più complesso di quanto non sia svolgere manualmente tale operazione, per diversi motivi.

Innanzitutto è difficile risalire al numero identificativo (*ID number*) che il modellatore assegna alle entità create nel caso occorresse applicare, per esempio, un carico nodale. Infatti per ogni elemento finito creato vengono generati almeno due nodi e per modelli composti da molti elementi prevedere quale sarà l'*ID* di un certo nodo diventa un lavoro improbo. Per questo motivo è preferibile lavorare sulla geometria, e se non è possibile fare altrimenti occorre avere una sorta di mappa dalla quale è possibile ricavare il numero identificativo dell'elemento nella posizione voluta.

Un'altra significativa complicazione deriva dalla necessità di generare una *mesh* con certe caratteristiche:

1. gli elementi componenti devono essere fatti in modo da evitare il più possibile eccessive distorsioni dalla forma del quadrilatero regolare, come si vede in figura B.1¹.

¹ Infatti i risultati di un'analisi tendono ad essere più accurati quando gli elementi sono compatti. Le distorsioni in figura di solito degradano l'accuratezza, in particolare risultano negative le *combinazioni* di più distorsioni. Il degrado riguarda soprattutto i campi di tensione piuttosto che gli spostamenti. Se le distorsioni sono inevitabili a causa della forma del modello, l'uso di elementi con nodi ai lati (ed anche interni) oltre che sugli angoli migliora l'accuratezza dell'analisi.

2. bisogna evitare assolutamente una situazione come quella in figura B.2, che comporterebbe il distacco delle due superfici una volta caricate (*effetto cerniera lampo*)

In virtù di queste necessità, si delineano alcuni problemi specifici:

- i lati degli elementi *shell* devono seguire l'andamento dei correnti ed avere i nodi corrispondenti con quelli degli elementi *bar* che rappresentano gli irrigidimenti dei pannelli e le flange dei longheroni
- le centine sono quadrangolari, poste tra due pannelli aventi spesso un diverso passo degli irrigidimenti ed in sostanza un diverso numero di nodi tra lato superiore ed inferiore, pertanto occorre gestire in qualche modo la transizione tra nodi dorsali e ventrali
- essendoci una certa rastremazione, alcuni correnti andranno ad interrompersi sul longherone anteriore; bisogna garantire la coerenza dei nodi nella zona di interruzione
- il numero di elementi di un *frame* (cioè la zona delimitata da due centine e due correnti) deve essere per quanto possibile sempre lo stesso e tale da evitare di avere elementi eccessivamente lunghi

Inutile dire che i requisiti elencati sono facilmente perseguibili se l'operatore corregge manualmente il modello, mentre risulteranno molto più difficili da realizzare nell'ambito di una procedura automatica.

Altro problema della procedura automatica si ha durante l'unione dei nodi di elementi contigui (*equivalence*). Questo comando impone a **Patran** di sostituire due nodi entro una certa distanza (tolleranza) con un unico nodo, saldando di fatto gli elementi che condividono questo nodo. Se la tolleranza scelta è troppo piccola i due nodi potrebbero rimanere separati e portare all'effetto cerniera lampo tra i due elementi, d'altra parte se la tolleranza fosse eccessivamente grande si rischierebbe di unire nodi che devono restare distinti con conseguente degenerazione (*collapse*) dell'elemento (figura B.3). Se nell'ambito di una procedura manuale è possibile rendersi conto con una serie di tentativi di quale sia la tolleranza corretta, in automatico occorre trovare un sistema che svolga l'operazione 'a colpo sicuro'.

È possibile individuare nella struttura di **PROGRAMMA.m** i blocchi di operazioni riassunti nel diagramma di flusso di figura B.4; ogni blocco logico verrà sviluppato nei paragrafi seguenti.

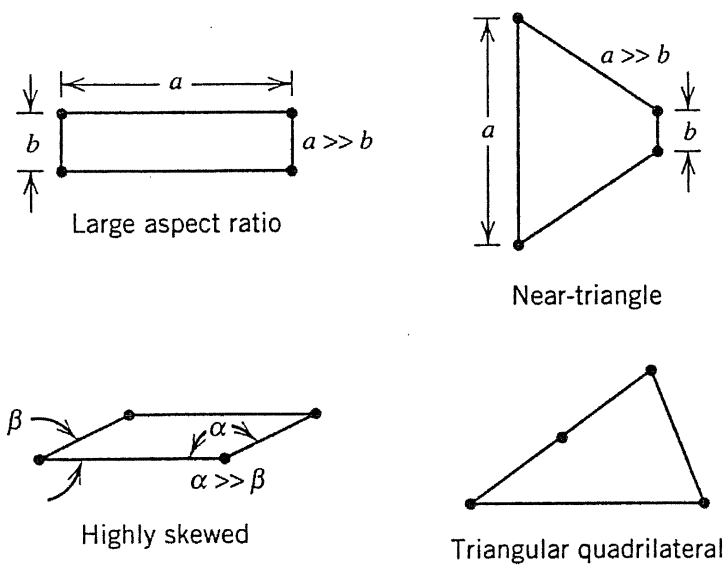


Figura B.1: Esempi di elementi distorti

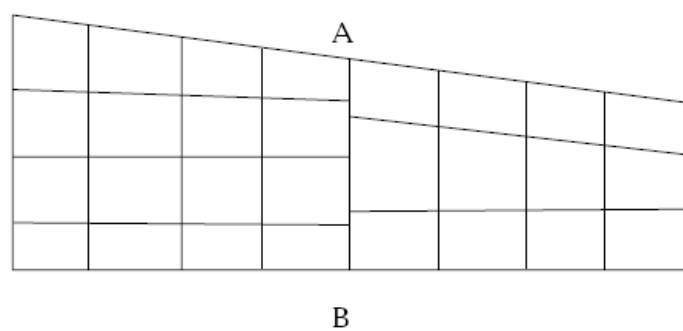


Figura B.2: Effetto cerniera lampo

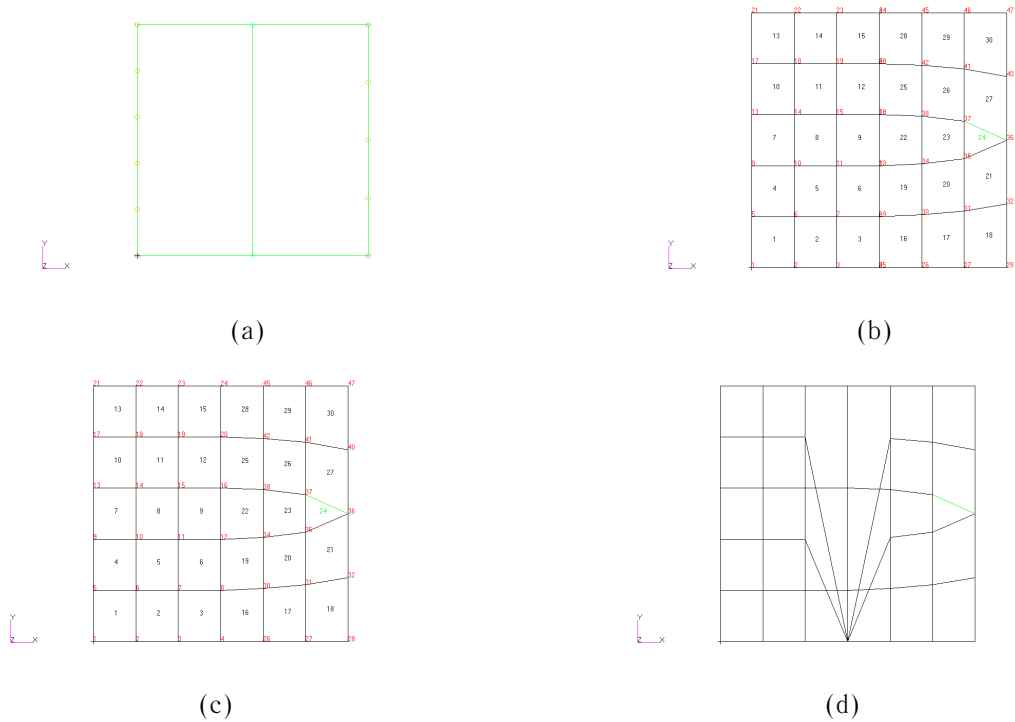


Figura B.3: Effetto del comando di *equivalence*: a) geometria iniziale ; b) *mesh* delle due superfici (con nodi all'interfaccia coincidenti) ; c) *Equivalence* con tolleranza corretta ; d) *Equivalence* con tolleranza eccessiva ed elementi collassati

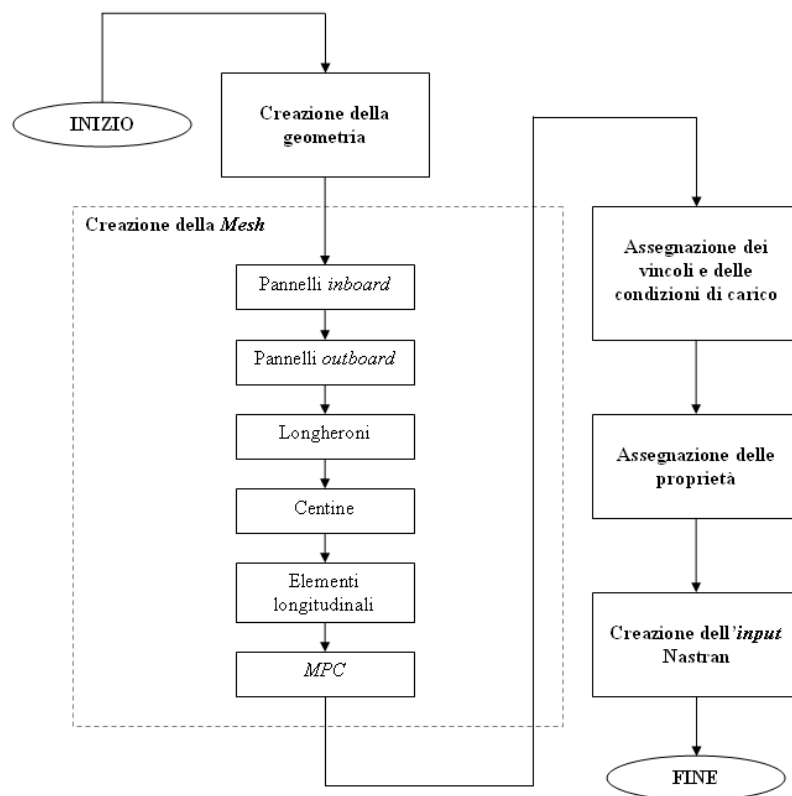


Figura B.4: Diagramma di flusso di **PROGRAMMA.m**

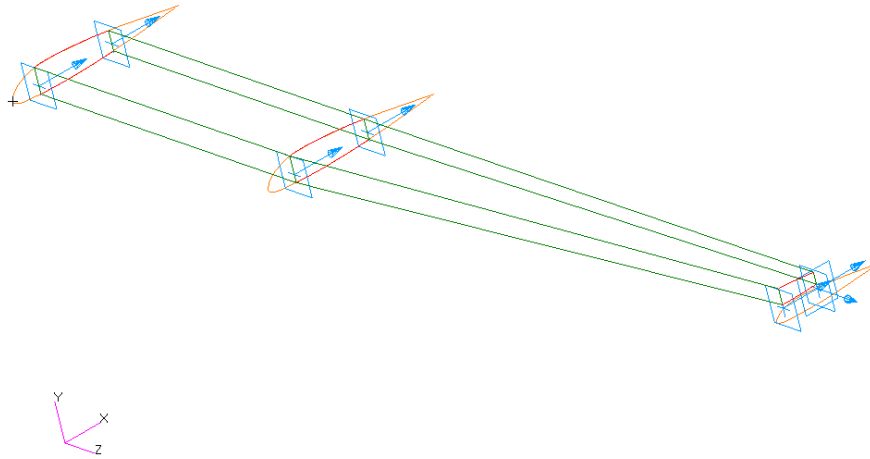


Figura B.5: Creazione della geometria esterna del cassone

B.1 Creazione della geometria

Nella prima parte del codice si procede all'acquisizione delle coordinate dei punti che giacciono sulla curva del profilo. Queste coordinate (ascissa ed ordinata), riportate su due colonne in dei *file* di testo relativi ai profili alla radice, *kink* ed estremità, vengono trasformate in vettori ed in seguito usati per eseguire il comando *Create* \Rightarrow *Curve* \Rightarrow *Spline* il quale permette di creare una curva dall'interpolazione di una lista di punti forniti sottoforma di terzetti di coordinate. È sufficiente aggiungere le coordinate precedentemente acquisite per generare il profilo, ed in questo modo vengono prodotte le tre curve **arancio** in figura B.5. Successivamente vengono creati i punti che individuano i vertici del cassone alare nella vista in pianta, le superfici esterne dell'ala col comando *Create* \Rightarrow *Surface* \Rightarrow *Curve* e l'opzione *2 Curve* ed i primi piani di taglio in corrispondenza dei vertici (**celeste**). Poiché il cassone è più corto della superficie alare, il profilo all'estremità viene ricavato dalla superficie esterna dell'ala col comando *Create* \Rightarrow *Curve* \Rightarrow *Intersect* che fa intersecare la superficie con una piano trasversale in prossimità del *tip* del cassone. A questo punto si tagliano le curve dei profili con *Edit* \Rightarrow *Curve* \Rightarrow *Break* ottenendo le porzioni di queste appartenenti al cassone (**rosso**), dalle quali è semplice creare tutte le superfici esterne (**verde**).

Sulle superfici del cassone vengono successivamente prodotte delle curve, che torneranno utili più avanti, che individuino ciascun corrente e centina del modello. Questa operazione si svolge in tre fasi: si generano dei punti in corda ed in apertura che corrispondano alla po-

sizione rispettivamente dei correnti e delle centine , in seguito da questi punti si formano dei piani, coi quali si individueranno le curve con il comando di intersezione. Per i punti si parte dai vertici al *root* e *kink* in corrispondenza del longherone posteriore e si traslano con *Transform* \Rightarrow *Point* \Rightarrow *Translate*, usando di volta in volta il passo delle centine (*inboard* ed *outboard*) o dei correnti (dorsale e ventrale) ed impostando una ripetizione (*Repeat Count*) di ciascuna traslazione congrua col numero di centine e di correnti.

La numerazione dei piani e delle curve viene controllata in modo da avere intervalli di numeri diversi per zone diverse del modello. Ad esempio i piani dal 10 al 49 identificheranno i correnti ventrali oppure le curve dal 400 al 449 individuano le centine sul pannello superiore *inboard*. Questo sistema di allocazione predefinita dei numeri identificativi in base alla zona del modello permette di avere sugli stessi un certo controllo che aiuterà nelle fasi successive del processo, pertanto verrà adottato sistematicamente. Naturalmente il numero complessivo di elementi non è definito e probabilmente ne verranno usati meno di quelli predisposti dalla numerazione.

In figura B.6 si vede come appare il modello in seguito alla creazione dei piani e delle curve. Per comodità di visualizzazione, i gruppi di piani relativi ai correnti del dorso e del ventre sono stati creati uno alla radice l'altro all'estremità.

Un inconveniente dovuto al fatto di non avere una geometria fissa è che, per determinati valori del passo tra le centine b_{rib} , è possibile che la lunghezza in apertura dell'ultima baia prima del *kink* o dell'estremità sia troppo grande. Le centine del modello vengono infatti create a partire dalla radice e dal *kink* e posizionate verso l'esterno, ciascuna ad una distanza pari al passo tra di esse.

È facile convincersi del fatto che le ultime baie avranno lunghezza pari a

$$L_{ultima} = L_{cassone} - n_{tot,rib} \cdot b_{rib} .$$

Nel caso in cui si avesse $L_{ultima} \geq 1.5 \cdot b_{rib}$, il codice provvede ad aggiungere un'altra centina a metà strada tra l'ultima aggiunta e quella d'estremità (*inboard* oppure *outboard*).

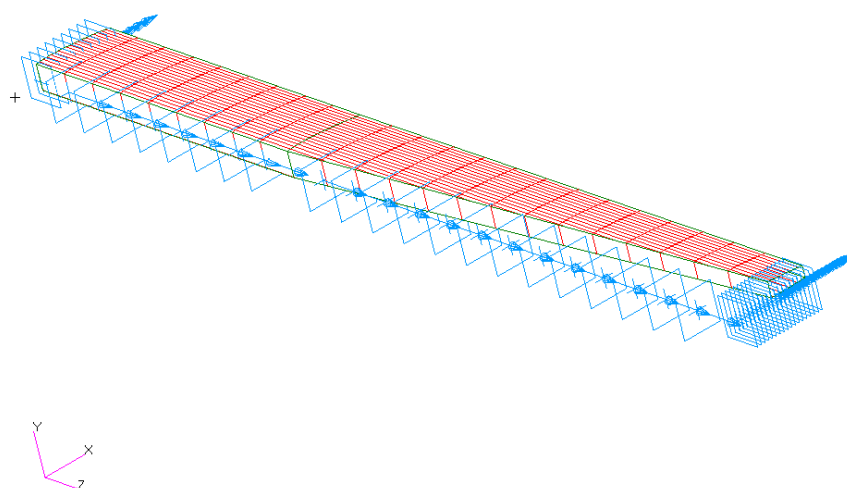


Figura B.6: Creazione dei piani e delle curve

B.2 Generazione della *mesh*

Pannelli

A parte la curvatura dovuta al profilo aerodinamico, i pannelli *inboard* sono semplicemente dei rettangoli sui quali si vuole ottenere una *mesh* regolare formata da elementi equispaziati in corda, escluso l'ultimo tra longherone anteriore e corrente, più corto, ed in apertura, esclusa l'ultima baia. Per ottenere questo si potrebbe dividere il rettangolo in quattro come mostrato in figura B.7(a): un rettangolo più grande con lati di lunghezza pari a multipli del passo tra i correnti e tra le centine, gli altri tre a completare la superficie del pannello. Mediante dei *mesh seed*, raffigurati in arancione, sui lati corti, imposti in modo da rispettare la spaziatura degli elementi longitudinali, ed un opportuno valore del *GEL* (vedi A.4) per la coerenza con i nodi delle centine sarebbe possibile ottenere un pannello con la *mesh* voluta.

In realtà, poichè le proprietà delle sezioni sono costanti baia per baia, è più utile suddividere il pannello in tante coppie di rettangoli quante sono le baie, in modo da poter assegnare in seguito le suddette agli elementi finiti associati alle superfici prodotte con questo procedimento. La geometria finale, compresi i *seed* in ogni baia, è riportata in figura B.7(b).

Nel codice è stato introdotto il parametro *rib_ar*, che determina il numero di elementi che verranno creati in un *frame*. Questo consente un certo controllo sull'allungamento dei componenti dei pannelli e sulla quantità totale degli stessi. Per una spaziatura delle centine nell'ordine dei $300 \div 500mm$ ed una spaziatura dei correnti di $100 \div 200mm$ circa un valore di 3 per il parametro consente di ottenere elementi ragionevolmente compatti. Se si dovesse utilizzare il *software* per l'ottimizzazione di un'ala diversa, potrebbe essere utile modificare il parametro.

Il valore di *rib_ar* viene usato per definire il *GEL* in modo da ottenere la *mesh* in figura B.7(c):

$$GEL = \frac{b_{rib}}{rib_ar}$$

Per i pannelli *outboard* vale lo stesso ragionamento. Infatti per l'attribuzione delle proprietà agli elementi, risulta più comodo spezzare la superficie trapezoidale originale in una coppia di superfici per ogni baia con lo stesso criterio adottato per i pannelli rettangolari, come si vede in figura B.8(a).

Il codice calcola baia per baia l'identificativo del corrente più vicino al longherone anteriore, passando per ogni *run out* all'irrigidimento successivo. Si è assunto che gli elementi longitudinali si interrompano in corrispondenza della centina, dunque per evitare che la *mesh* sia quella in figura B.8(b), nella quale si avrebbero i lati degli elementi **1** e **2** adiacenti al lato dell'elemento **3** ed il nodo **A** soggetto di conseguenza all'effetto cerniera lampo, si esegue il comando *Associate* \Rightarrow *Point* \Rightarrow *Surface* sul nodo e sulla superficie dell'elemento

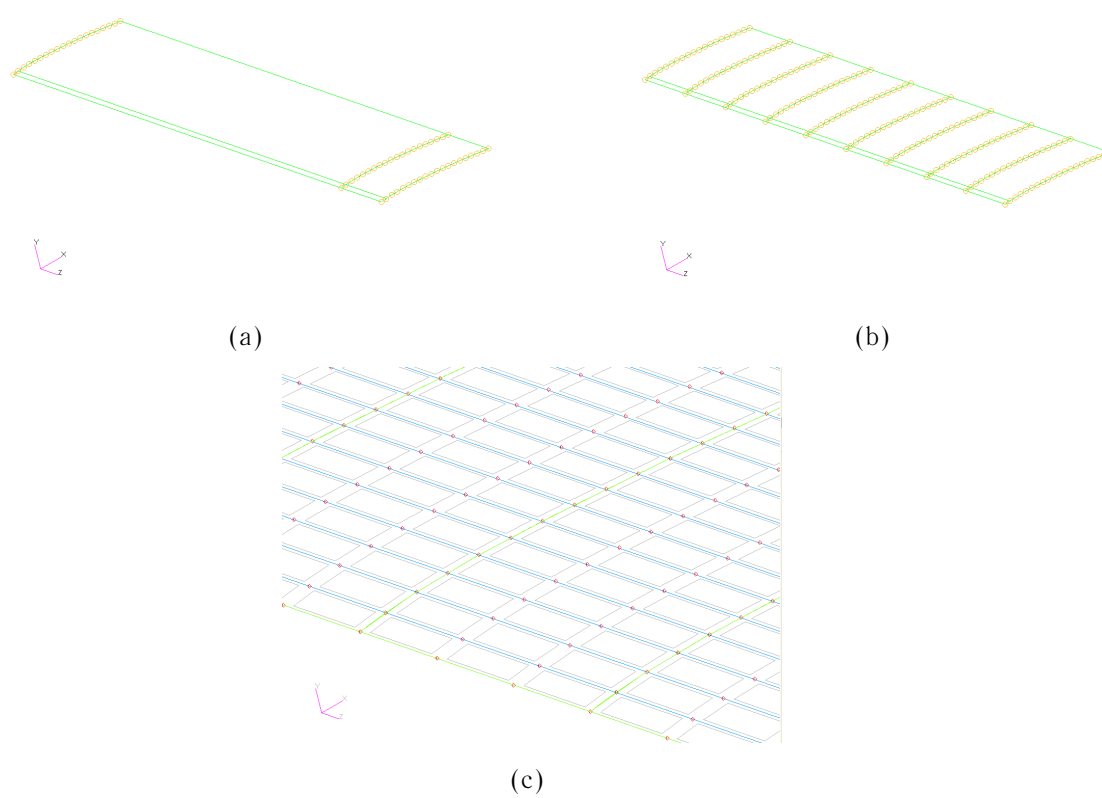


Figura B.7: Decomposizione del pannello *inboard*

3. In questo modo si forza la creazione di un nodo nel punto associato, con il risultato in figura B.8(c).

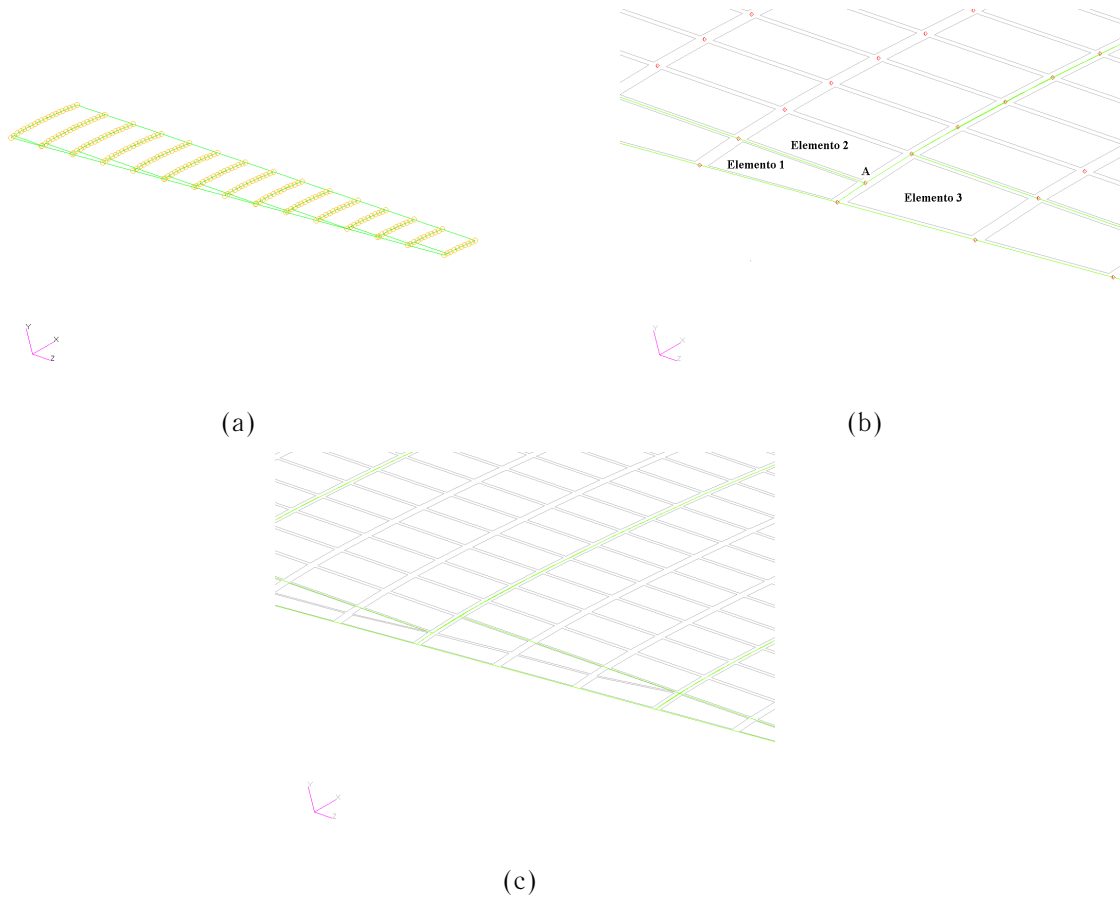


Figura B.8: Decomposizione del pannello *outboard*

Longheroni e centine

I longheroni sono formati da superfici rettangolari, non costituiscono dunque un grosso problema per quanto concerne la loro semplificazione geometrica. Comunque anche per queste parti si è adottata la decomposizione per ogni baia. È stato inoltre introdotto il parametro di proporzionamento *spar_ar*, che definisce il numero di elementi compresi tra dorso e ventre. Pertanto tra due setti il longherone sarà composto da $rib_ar \cdot spar_ar$ elementi.

Le centine invece richiedono un po' d'attenzione in più. Sostanzialmente i nodi sul perimetro sono già stati prodotti in virtù delle *mesh* realizzate sui pannelli e sui longheroni; per evitare l'effetto cerniera lampo occorre che la *mesh* di questi componenti abbia (quasi sempre) un numero di nodi tra dorso e ventre diverso (vedi B.9(a)). Se si richiedesse a **Patran**

di generare una *mesh* direttamente sulla superficie si otterrebbe il risultato in figura B.9(b), ritenuto insoddisfacente.

Allora si è proceduto in questo modo: la superficie di partenza è stata suddivisa in tre, una parte centrale e due striscie adiacenti ai due pannelli, come riportato in figura B.9(c). La parte centrale contiene un ugual numero di nodi nei lati superiore ed inferiore, definito dal parametro *rib_param* secondo la seguente relazione:

$$N_{nodi,centrale} = rib_param \cdot \frac{N_{nodi,dorso} + N_{nodi,ventre}}{2}$$

La parte centrale ha quindi un numero di nodi pari alla media tra i nodi sui due lati, multipli-

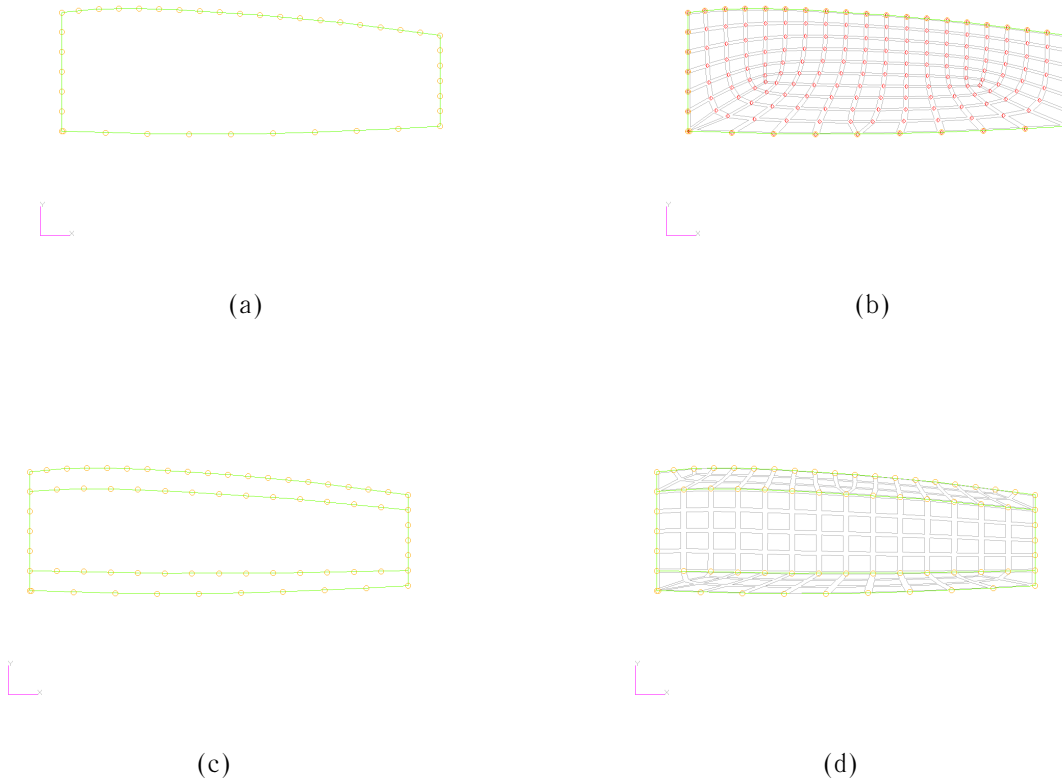


Figura B.9: *Mesh* delle centine

cata per il parametro di controllo. Il risultato finale dell'operazione è una *mesh* decisamente migliore, come si vede bene in figura B.9(d).

Elementi longitudinali ed *MPC*

Gli elementi longitudinali, ovvero irrigidimenti dei pannelli e flange dei longheroni, non costituiscono un particolare problema: a partire dalle curve create dall'intersezione tra piani dei correnti e superfici dei pannelli, che si estendono per tutto il cassone, si ottengono dei segmenti lunghi quanto le baie tagliando le curve con i piani delle centine. A questo punto si creano gli elementi *bar* sui segmenti, avendo cura di impostare per questi lo stesso *GEL* usato per i pannelli ed i longheroni.

Per gli *MPC* che introducono i carichi sul modello ci si è scontrati con l'inconveniente anticipato all'inizio di questa appendice. Infatti per creare un *RBE3* occorre fornire una lista di nodi indipendenti, giacenti in questo caso sul perimetro della centina, ed uno dipendente nel quale verrà applicato il carico. Sebbene il punto d'introduzione non esista ancora, i nodi indipendenti sul perimetro ci sarebbero, generati in fase di creazione dei pannelli e dei longheroni. Purtroppo non è facile risalire agli *ID* di questi nodi, pertanto se ne creano altri coincidenti con quelli sul perimetro col comando *Create* \Rightarrow *Node* \Rightarrow *Extract* e l'opzione *Curve*, fornendo di volta in volta la posizione parametrica voluta. Ad esempio sui lati relativi ai longheroni le posizioni sono le seguenti:

$$\frac{1}{spar_ar}, \frac{2}{spar_ar}, \dots, \frac{spar_ar - 1}{spar_ar}$$

mentre sui lati relativi ai pannelli si opera allo stesso modo, con l'accorgimento ulteriore di tener conto della distanza dell'ultimo corrente dal longherone anteriore, diversa dalla spaziatura fra gli irrigidimenti. Le entità prodotte in questo modo avranno numeri identificativi **noti**.

Il nodo dipendente viene creato in modo analogo, usando ancora il comando *Create* \Rightarrow *Node* \Rightarrow *Extract*, stavolta con l'opzione *Single node on surface* e fornendo la posizione parametrica del baricentro geometrico della superficie dalla centina : $u = 0.5, v = 0.5$. Questo rappresenta la traccia dell'asse elastico sul piano del setto. Niente vieta di prendere una posizione diversa (ad esempio il baricentro di massa della sezione), ma esulerebbe dal carattere dimostrativo del presente lavoro.

A questo punto si ha una serie di *MPC* in corrispondenza delle centine e scollegati dal resto del modello. Sarà sufficiente però eseguire un *Equivalence* (vedi più avanti) per attaccarli alla struttura senza problemi.

Unione dei nodi vicini (*Equivalence*)

Poiché il modello realizzato secondo le indicazioni esposte finora è composto da superfici affacciate l'una sull'altra e sulle quali sono state costruite delle *mesh* separate, avranno

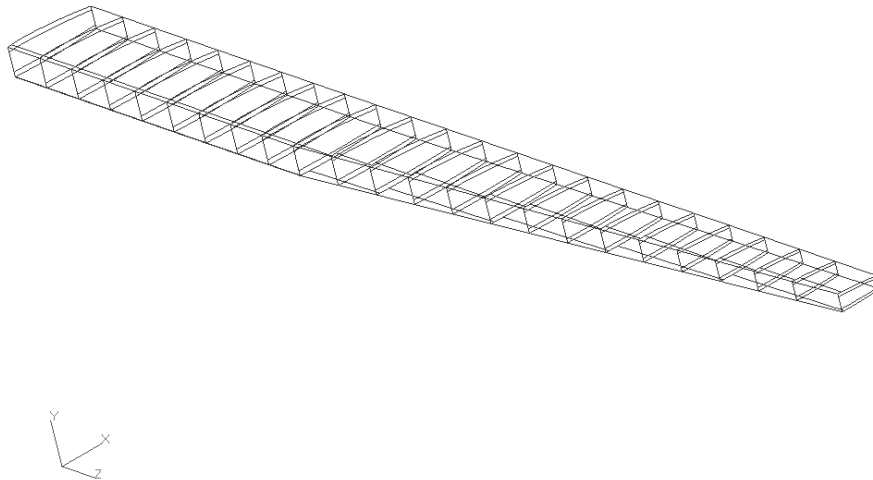


Figura B.10: Modello 'sconnesso'

nei lati comuni delle coppie di nodi coincidenti, appartenenti ciascuno ad una diversa superficie. Se si lasciasse il modello così, sarebbe considerato dal solutore come un insieme di pezzi sconnessi, rendendo di fatto impossibile l'analisi. Si può facilmente dimostrare la non integrità del modello con il comando *Verify* \Rightarrow *Element* \Rightarrow *Boundaries*, che fornisce il risultato di figura B.10.

Dapprima si stabilisce, in base alle dimensioni caratteristiche del modello una tolleranza piuttosto larga, *toll_equiv*. Il preprocessore d'altro canto stima una tolleranza *critica*, *toll_crit*, tale per cui un *equivalence* dato con tolleranza maggiore di *toll_crit* causerebbe il collasso di almeno un elemento. Si dà quindi il comando *Equivalence* \Rightarrow *All* \Rightarrow *Tolerance cube* con tolleranza pari a *toll_equiv* e l'opzione *Allow tolerance reduction*. In questo modo **Patran** confronta la tolleranza fornita con la critica, e se la prima fosse maggiore della seconda la ridurrebbe automaticamente al valore critico ed opererebbe l'*equivalence* col valore ridotto.

Per motivi legati probabilmente alla diversa curvatura dei pannelli e delle curve giacenti su di essi, questa operazione da sola non è sufficiente ad attaccare gli elementi *bar* ai pannelli. Allora si fornisce il comando *Equivalence* \Rightarrow *Group* \Rightarrow *Tolerance cube*, circoscritto ai soli gruppi che contengono gli elementi dei pannelli e dei correnti e con l'opzione *Allow collapsed elements*. Con questo secondo comando più 'forte', si riesce ad attaccare gli

elementi longitudinali allo *skin*. Se il valore della tolleranza è stato valutato correttamente non ci dovrebbe essere il rischio di far degenerare qualche elemento.

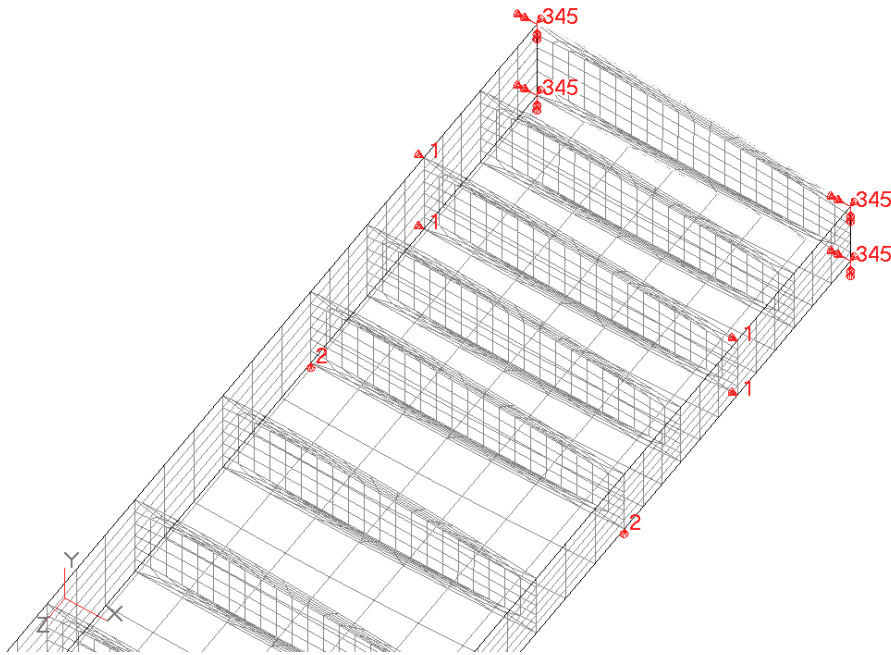


Figura B.11: Modello vincolato

B.3 Assegnazione dei vincoli e dei carichi

L'ala di riferimento non attraversa la fusoliera ed è collegata ad essa mediante due staffe e due pendoli per semiala, le prime, in corrispondenza della seconda centina, a prevenire gli spostamenti nella direzione del vento; le seconde, che forniscono un vincolo alla traslazione verticale della struttura, in corrispondenza dei longheroni e della quarta centina. Il quadro è completato dalla simmetria all'interfaccia tra le due semiali, che impedisce le traslazioni nella direzione dell'apertura e le rotazioni nel piano latero direzionale.

I vincoli sono assegnati col comando *Create* \Rightarrow *Displacement* \Rightarrow *Nodal* alla geometria del modello. In figura B.11 è possibile vedere i vincoli sul modello.

I carichi vengono introdotti nel modello dai nodi dipendenti degli *MPC* creati precedentemente. Prevedendo la necessità di avere varie condizioni di volo con carichi baia per baia diversi, si forniscono delle sollecitazioni - taglio, momento flettente e torcente - unitarie in ogni nodo degli *MPC*, dopodiché vengono create le condizioni di ogni caso studiato moltiplicando questi carichi unitari al valore effettivo delle sollecitazioni per la condizione valutata di volta in volta. È possibile vedere il modello caricato in figura B.12.

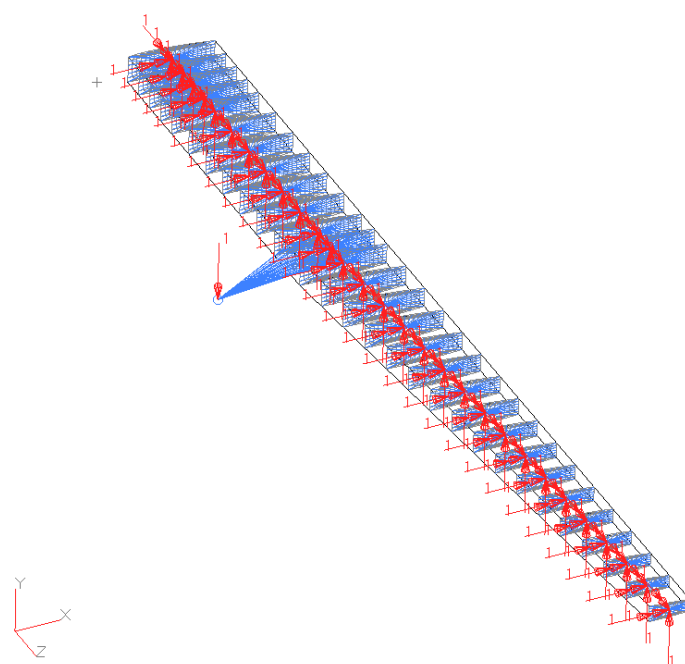


Figura B.12: Modello caricato

B.4 Attribuzione delle proprietà e creazione dell'*input* Nastran

Una volta fornite le caratteristiche del materiale, con i comandi *Create* \Rightarrow *2D* \Rightarrow *Shell* e *Create* \Rightarrow *1D* \Rightarrow *Beam* si assegnano le proprietà agli elementi del modello. Le dimensioni geometriche della sezione, costanti lungo la baia, vengono calcolate da una *routine* che le ricava per interpolazione lineare dei valori in tre stazioni dell'apertura. Si procede baia dopo baia assegnando una proprietà per ogni pannello (rivestimento dorsale o ventrale, anima dei longheroni, centina) e per ogni elemento longitudinale (irrigidimento o flangia). A questo punto viene richiesto un .rpt nel quale **Patran** scrive il peso della struttura.

Successivamente un'altra *routine* calcola delle matrici che contengono gli *ID number* di ogni elemento dei pannelli e dei longheroni. Infatti l'intero processo di creazione della *mesh* è stato pianificato in modo che gli elementi venissero formati secondo una precisa sequenza che è stato possibile riprodurre, secondo uno schema fisso, rappresentato a titolo d'esempio in figura B.13 per il pannello dorsale *inboard*.

Conoscere l'identificativo di ogni elemento è utile per creare i gruppi di elementi per le verifiche. Infatti (si veda l'appendice C) sul modello sono verificate le tensioni sugli elementi nella mezzeria di ogni baia, che vengono per comodità raggruppati, distinti fra quelli del pannello dorsale e ventrale e quelli sulle anime. In figura B.14 vi è un dettaglio della vista dei gruppi soluzione, che sembrano delle 'fascette' avvolte al cassone.

Alla fine del lungo processo di creazione del modello, esso è pronto per essere sottoposto all'analisi strutturale con **Nastran**, che ha bisogno in ingresso del .bdf che contiene tutte le informazioni sul modello necessarie per il calcolo. La scrittura di questo *file* è l'ultimo comando eseguito da **PROGRAMMA.m**.

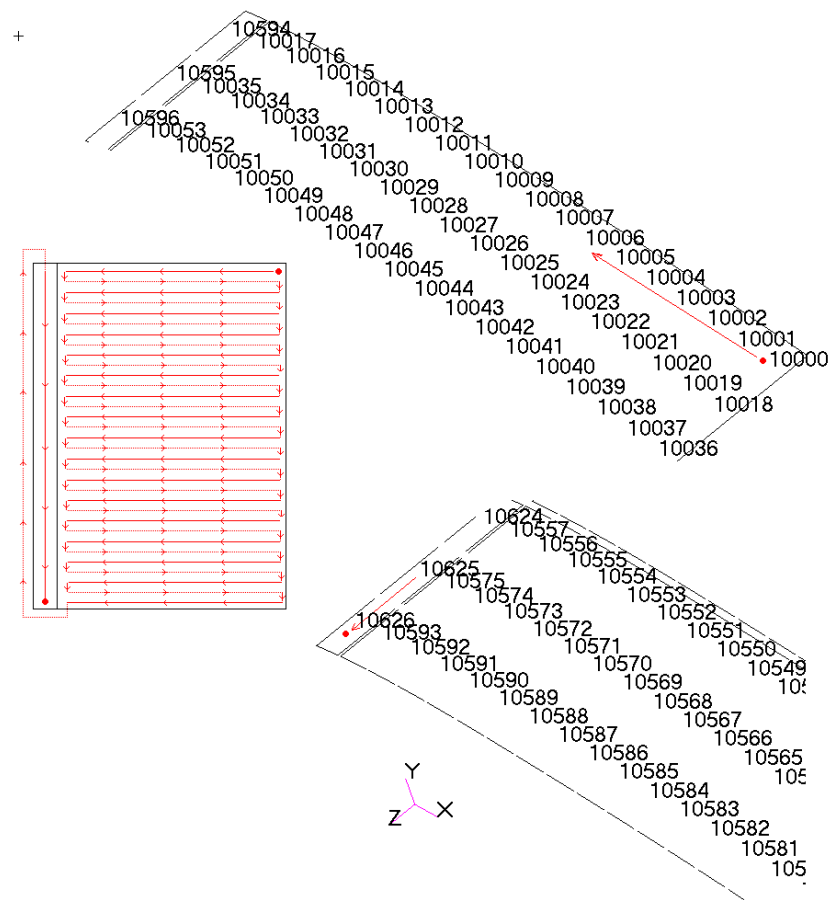


Figura B.13: Schema della numerazione degli elementi

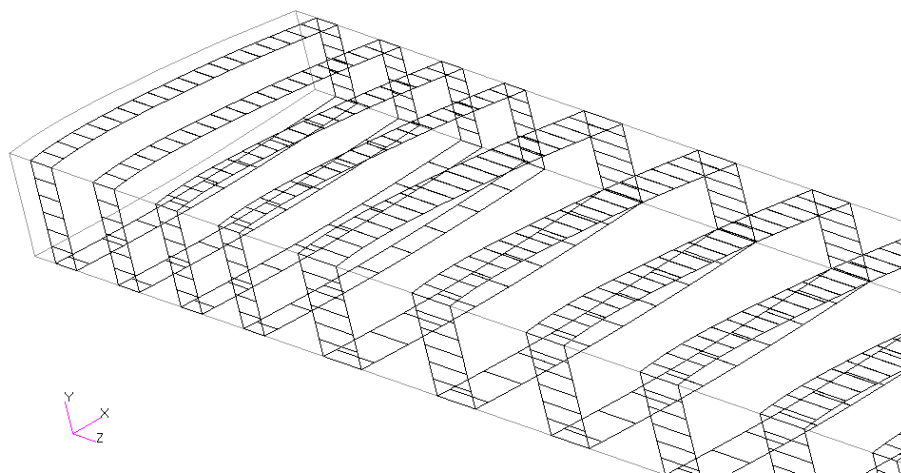


Figura B.14: Fascette di gruppi soluzione

APPENDICE C

IL PROCESSO DI VERIFICA DELLE DIMENSIONI

In questa appendice si illustreranno le funzionalità del codice **RISULTATI.m**, che esegue le operazioni di analisi post-processo del modello, insieme con **VETTORI.m**. A differenza di quanto si è dovuto fare per la fase pre-processo con **PROGRAMMA.m** (vedi appendice B), il compito di raccogliere i dati elaborati per eseguirne la verifica e riorganizzare la mole di informazioni in maniera da essere di più facile consultazione è stato decisamente meno gravoso. Il codice qui illustrato entra in gioco nel momento in cui, all'interno di un ciclo di ottimizzazione, il modello del cassone definito dall'*input* di *modeFrontier* è stato generato e sottoposto all'analisi con **NASTRAN**.

Si parte dunque da un *.db* del modello ed un *.xdb* dell'analisi, l'intenzione è di scrivere un *file* di sessione **Patran** che riferisca l'uscita di **NASTRAN** al modello (operazione di *attachment* del *.xdb* al *.db*) e successivamente estrarre nella maniera più semplice possibile i dati necessari per le verifiche. Per fare ciò si distinguono due possibili strategie: gestire la fase post-processo soltanto con **Patran** (attraverso la sessione creata) e *modeFrontier*, oppure fornire in ingresso ad un altro codice **Matlab** i dati della soluzione agli elementi finiti. Si è preferito intraprendere, con il codice **VETTORI.m**, la seconda strada, ritenuta migliore sia per la semplificazione dello schema *mF* (alleggerito di alcune variabili), sia per la possibilità di gestire meglio il flusso di dati e di aggiungere miglioramenti. A titolo di esempio, si ritiene molto più semplice calcolare con **MATLAB** i fattori di correzione per la plasticità del carico critico a compressione in quanto dipendenti dalla tensione esercitata sulla parte, nota soltanto dopo che è stato effettuato l'*attachment*.

È stato invece possibile mediante **VETTORI.m** stampare per ogni configurazione dei diagrammi come quelli in figura C.1, utili per avere un riscontro visivo immediato delle sollecitazioni in apertura e della bontà della soluzione, indicata da quanto è distante l'involuppo dei margini di sicurezza in apertura dalla linea dello zero. Infatti in linea di principio una solu-

zione di ottimo dovrebbe avere i margini di sicurezza in tutte le baie per tutte le sollecitazioni il più vicino possibile ad annullarsi, segno che tutte le forme di collasso avvengono contemporaneamente ed il materiale viene sfruttato completamente senza sovradimensionamenti, che ridurrebbero l'efficienza strutturale.

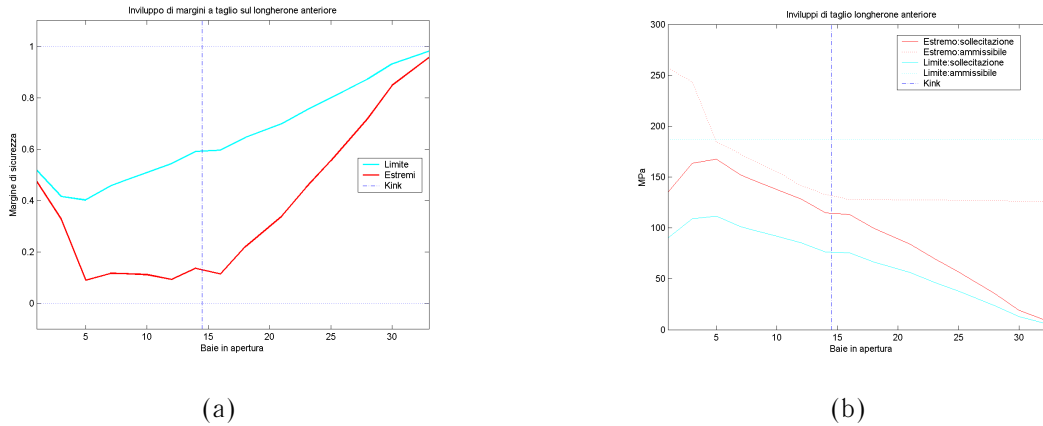


Figura C.1: Esempio di diagrammi in apertura: a) margini di sicurezza ; b) sollecitazione

Noto il numero di baie da verificare, assegnato dall'operatore tramite una variabile di ingresso nell'ambiente dell'ottimizzatore, **RISULTATI.m** calcola in quali baie effettuare le verifiche. Non è possibile assegnare dei numeri di baia a priori, perché potrebbero riguardare una stazione prima o dopo del *kink* o per certe configurazioni non esistere, dato che al variare della spaziatura delle centine ogni configurazione ha un diverso numero totale di baie. Con un semplice algoritmo di arrotondamento viene valutato un insieme di possibili stazioni in apertura, in modo da avere punti di controllo più o meno alla stessa posizione percentuale in apertura. Per intendersi, per un cassone con 10 baie da controllare in 4 stazioni, i punti di controllo saranno alle baie 2, 5, 7 e 10, un cassone con 20 baie verrà verificato invece alle baie 4, 10, 14, 20, ma in entrambi i casi i punti di controllo cadranno al 25% in apertura, al 50%, al 75% ed al 100%. L'estrazione dei dati viene svolta da *mF* mediante la lettura di una serie di *report*, uno per ogni condizione di volo, ciascuno contenente i risultati relativi a tutte le baie da controllare, *report* richiesti dalla sessione scritta da **RISULTATI.m**.

Successivamente **VETTORI.m** riceve in ingresso i dati estratti e li rielabora, calcolando delle matrici, una per ogni condizione di carico, aventi tante righe quante sono le stazioni da controllare e tante colonne quante sono le sollecitazioni analizzate. Allo stesso modo, vengono create analoghe matrici con gli ammissibili ed i margini di sicurezza. I vettori delle condizioni da verificare vengono passati a *mF*, che impone sostanzialmente il vincolo che i margini di sicurezza siano tutti quanti positivi e l'obiettivo di minimizzarne il valore assoluto, in modo da avere soluzioni che si avvicinino alla situazione auspicata dall'assioma del Gerard.

APPENDICE D

LA CARD NLPARM

NLPARM	ID	NINC	DT	KMETHOD	KSTEP	MAXITER	CONV	INTOUT	
NLPARM		20	0.	AUTO	5	25	PW	NO	+NP1
	EPSU	EPSP	EPSW	MAXDIV	MAXQN	MAXLS	FSTRESS	LSTOL	
+NP1	10^{-3}	10^{-3}	10^{-7}	3	MAXITER	4	0.2	0.5	+NP2
		MAXBIS				MAXR		RTOLB	
+NP2		5				20.		20.	

Tabella D.1: Ingressi della *card* NLPARM

Quando il preprocessore scrive il *file* .bdf per richiamare l'analisi di *NASTRAN* nel caso di *SOL106*, nel testo comprende anche tre righe al massimo, introdotte dal comando **NLPARM**, che determinano il modo con cui il solutore deve affrontare il problema non lineare (figura D.1).

In tabella D.1 è riportato lo schema completo, tratto da (4), degli ingressi della *card* con alcuni valori di *default*.

ID numero intero progressivo riferito al *load case*

NINC numero di incrementi di carico: determina in quanti *step* di incremento verrà suddiviso il ciclo di carico, secondo la relazione $\Delta P = \frac{P_i - P_{i-1}}{NINC}$ ed essendo P_i e P_{i-1} i carichi del *load case* attuale e precedente

KMETHOD definisce la strategia di aggiornamento della matrice di rigidezza K :

AUTO determina automaticamente, in base ad una stima del tempo di calcolo richiesto per svolgere il ciclo con e senza aggiornamento di K , quando aggiornare

```

ECHO = NONE
$ Direct Text Input for Global Case Control Data
SUBCASE 1
$ Subcase name : Carico_unitario_x200
  SUBTITLE=Carico_unitario_x200
  NLPARM = 1
  SFC = 2
  LOAD = 2
  DISPLACEMENT(SORT1,REAL)=ALL
  SPCFORCES(SORT1,REAL)=ALL
  STRESS(SORT1,REAL,VONMISES,BILIN)=ALL
$ Direct Text Input for this Subcase
BEGIN BULK
PARAM POST 0
PARAM AUTOSPC NO
PARAM LGDISP 1
PARAM PRTMAXIM YES
NLPARM 1 10 AUTO 5 25 YES
$ Direct Text Input for Bulk Data
$ Elements and Element Properties for region : p
PSHELL 1 1 1 1
$ Pset: "p" will be imported as: "pshell.1"
CQUAD4 1 1 1 2 8 7
CQUAD4 2 1 2 3 9 8
CQUAD4 3 1 3 4 10 9
CQUAD4 4 1 4 5 11 10
CQUAD4 5 1 5 6 12 11
CQUAD4 6 1 7 8 14 13
CQUAD4 7 1 8 9 15 14

```

Figura D.1: Riga di comando nell'*input* del solutore che richiama la sequenza non lineare

SEMI funziona come AUTO, ma aggiorna obbligatoriamente alla seconda iterazione
ITER aggiorna ad intervalli regolari

KSTEP è il numero massimo di iterazioni che possono svolgersi senza aggiornamento di K ;
 se abbinato a ITER, aggiorna ogni KSTEP iterazioni, se abbinato ad AUTO o SEMI
 , aggiorna **al massimo** ogni KSTEP iterazioni.

Ad esempio, $KMETHOD = ITER$ e $KSTEP = 1$ è il metodo di N-R classico,
 mentre $KSTEP > MAXITER$ è il N-R modificato

MAXITER Numero massimo di iterazioni consentito per ogni ΔP di carico. Se *NASTRAN*, dopo
 aver compiuto MAXITER iterazioni, non ha raggiunto una soluzione accettabile, cioè
 entro le tolleranze richieste:

1. effettuerà un altro *step* con incremento di carico dimezzato (**bisezione**, si veda più sotto)
2. sceglierà la soluzione migliore, ovvero quella con errore più basso tra tutte quelle effettuate

La scelta dell'azione da eseguire viene fatta in base all'evoluzione dell'analisi ed ai valori di MAXBIS e MAXDIV

CONV, EPSU, EPSP, EPSW definiscono che tipo di test di convergenza realizzare ad ogni *step* (CONV) e le relative tolleranze richieste (EPSU, EPSP, EPSW); tutte le condizioni devono essere rispettate affinché sia conseguita la convergenza

INTOUT NO riporta le uscite richieste soltanto per l'ultimo *step* di carico

YES riporta le uscite richieste per tutti gli *step* di carico

MAXDIV, MAXBIS controllano il comportamento del ciclo iterativo al raggiungimento del limite di iterazioni o quando si è in presenza di divergenza.

MAXDIV determina il numero massimo di probabili divergenze nel processo prima che la soluzione venga ritrattata, cioè ricalcolata con la rigidità dello *step* precedente; se il valore viene superato due volte non consecutive prende la soluzione migliore (*best attainable*) e passa all'incremento successivo, se viene superato due volte consecutive interrompe il calcolo con un messaggio di errore.

MAXBIS determina il numero massimo di bisezioni dell'incremento di carico, superato il quale prende la soluzione migliore e passa allo *step* successivo; le cause possibili di una bisezione sono per esempio il raggiungimento di MAXITER iterazioni o la divergenza della soluzione.

MAXQN, MAXLS, LSTOL controllano il funzionamento degli algoritmi, che migliorano la stabilità e l'efficienza del processo di calcolo, *line search* e *Quasi-Newton*.

DT incremento di tempo per ogni *step*, nel caso di analisi dipendente dal tempo

FSTRESS controlla il calcolo relativo alla non linearità del materiale

MAXR controlla la modifica della lunghezza d'arco allo step attuale rispetto alla lunghezza al primo step: $\frac{1}{MAXR} \leq \frac{\Delta l_n}{\Delta l_0} \leq MAXR$

RTOLB massima rotazione incrementale consentita senza che si attivi la bisezione dell'incremento di carico

BIBLIOGRAFIA

- [1] *MSC Patran 2005 PCL and Customization manual.*
- [2] *MSC.Nastran 2005 Linear Static Analysis User's Guide.*
- [3] *MSC.Nastran 2005 Reference Manual.*
- [4] *NX Nastran, Handbook of Nonlinear Analysis (SOL 106).*
- [5] Ira H. Abbott e Albert E. Von Doenhoff. *THEORY OF WING SECTIONS.* DOVER PUBLICATIONS INC., 1959.
- [7] Herbert Becker. *Handbook of structural stability, part ii - buckling of composite elements.* Relazione Tecnica 3782, NACA TN.
- [8] Martina Bitossi, Attilio Salvetti, e Daniele Fanteria. *Sviluppo di metodi per la progettazione di strutture alari: validazione mediante analisi ad elementi finiti.* UNIVERSITÀ DEGLI STUDI DI PISA, a.a. 2006/2007.
- [9] E.F. Bruhn. *Analysis and Design of Flight Vehicle Structure.* Jacobs Publishers, 1975.
- [10] Robert D. Cook. *Finite element modeling for stress analysis.* John Wiley & Sons Inc., 1995.
- [11] Gianni Duccini, Daniele Fanteria, e Luigi Lazzeri. *Strumenti e metodi di progetto innovativi per strutture aeronautiche in materiale composito.* UNIVERSITÀ DEGLI STUDI DI PISA, a.a. 2006/2007.
- [12] EASA. *Certification Specifications for Large Aeroplanes CS-25,* 2006.

- [13] Robert M. Jones. *Mechanics of composite materials*. Taylor & Francis, 1999.
- [14] J.B. De Jonge. An introductory course on aircraft loading.
- [16] Paul Kuhn, James P. Peterson, e L. Ross Levin. A summary of diagonal tension, part i: Methods of analysis. Relazione Tecnica 2661, NACA.
- [17] Paul Kuhn, James P. Peterson, e L. Ross Levin. A summary of diagonal tension, part ii: Experimental evidence. Relazione Tecnica 2662, NACA.
- [18] Ted L. Lomax. *Structural load analysis for commercial transport aircraft: theory and practice*. AIAA EDUCATION SERIES, 1996.
- [19] Micheal C.Y. Niu. *AIRFRAME structural design*. Lockheed Aeronautical Systems Company Burbank, California, 1988.
- [20] Micheal C.Y. Niu. *Composite airframe structures, practical design information and data*. CONMILIT PRESS LTD., 1992.
- [21] Micheal C.Y. Niu. *AIRFRAME stress analysis and sizing*. HONG KONG CONMILIT PRESS LTD., 1999.
- [22] Daniel P. Raymer. *Aircraft Design: A Conceptual Approach*. AIAA EDUCATION SERIES, 1992.
- [23] Robert M. Rivello. *Theory and Analysis of Flight Structures*. McGraw-Hill, 1969.
- [24] A. Salvetti e D. Fanteria. Esercitazioni di costruzioni aeronautiche.
- [25] Klaus Otto Schwarzmeier. Diag2 - geometric non-linear parameters calculation for diagonal tension simulation using msc/nastran. Relazione tecnica, EMBRAER.

ELENCO DELLE FIGURE

2.1	Ala del velivolo di riferimento	4
2.2	Superficie e cassone alare di riferimento	5
2.3	Modello Patran della superficie alare	7
2.4	Modello Patran del cassone alare	8
2.5	Sezione tipica del cassone	9
2.6	<i>Run out</i> dei correnti : dettaglio	9
2.7	Esempio di errore nella modellazione del <i>run out</i>	10
2.8	Le centine nel modello <i>FE</i>	11
2.9	<i>Mesh</i> delle centine: caso in cui le spaziature dei correnti sono uguali	11
2.10	<i>Mesh</i> delle centine, caso con spaziature molto diverse tra dorso e ventre	12
2.11	<i>Mesh</i> delle centine, <i>rib_param</i> = 0.5	12
2.12	Vincoli sul modello	13
2.13	Introduzione dei carichi aerodinamici ed inerziali: <i>MPC RBE3</i>	14
2.14	Dettaglio di un <i>RBE3</i>	15
2.15	<i>MPC</i> che rappresenta l'attacco del motore	15
2.16	Forze agenti sul profilo: risultante verticale e momento aerodinamici (<i>azzurro</i>), distribuzione di massa strutturale (<i>nero</i>) e distribuzione di massa del combustibile (<i>rosso</i>)	16
2.17	Carichi unitari agenti sulle centine del modello	16
2.18	Modello <i>FE</i> completo: <i>wireframe</i>	17
2.19	Modello <i>FE</i> completo: <i>hidden line</i>	18
2.20	Modello <i>FE</i> completo: <i>shaded</i>	19
3.1	Tipico diagramma di manovra di un velivolo da trasporto civile	23
3.2	Effetto della comprimibilità sulla curva di stallo	25

3.3	Velocità della raffica di riferimento	26
3.4	Forze agenti sul profilo	31
3.5	Andamento delle tensioni in funzione delle forze applicate	32
3.6	Forma in pianta equivalente per l'applicazione del metodo del Diederich	33
3.7	Superfici tra centine, semibaie diverse (\hat{y}_1) e uguali (\hat{y}_2)	34
3.8	Calcolo dei momenti di trasporto	35
3.9	Gruppi di soluzione del ciclo	36
3.10	Instabilità iniziale del pannello irrigidito	38
3.11	Diagramma K/K_0	38
3.12	Instabilità globale del pannello irrigidito	41
3.13	Coefficiente di vincolo della piastra caricata a taglio	42
4.1	Schema semplificato di un progetto <i>modeFrontier</i>	46
4.2	Diagramma di flusso dell'ottimizzatore	48
4.3	Diagramma illustrativo del metodo di inserimento delle dimensioni rastremate	51
4.4	Schema completo del progetto <i>modeFrontier</i>	52
4.5	Esempio di valutazione dei margini di sicurezza	54
5.1	Ottimizzazione del passo dei correnti: peso del cassone (a); passo dei correnti dorsali (b)	60
5.2	Ottimizzazione del passo dei correnti: freccia massima percentuale (a); <i>scatter</i> tra peso e freccia (b)	61
5.3	<i>Scatter</i> tra peso e freccia: rappresentazione con bolle sfumate	62
5.4	Percentuali di violazione dei vincoli	62
5.5	Andamento delle peso: dettaglio delle ultime configurazioni	63
5.6	Confronto fra caratteristiche lungo l'apertura nelle due configurazioni: spessori (a) ed aree (c) della soluzione più leggera, spessori (b) ed aree (d) della più rigida	66
5.7	Andamento degli involuppi lungo l'apertura per la soluzione più leggera: margini di tutte le sollecitazioni (a); margini di taglio estremo sul longherone anteriore (c); taglio limite ed estremo (e); per la soluzione più rigida: margini di tutte le sollecitazioni (b); margini di taglio estremo sul longherone anteriore (d); taglio limite ed estremo (f)	67
5.8	Soluzione leggera: deformazione nella condizione di manovra a V_A e peso al decollo, scala della deformazione 1:1	68
5.9	Soluzione rigida: deformazione nella condizione di manovra a V_A e peso al decollo, scala della deformazione 1:1	68

5.10	Soluzione leggera: tensioni longitudinali nei pannelli: dorsale (a); ventrale (b)	69
5.11	Soluzione rigida: tensioni longitudinali nei pannelli: dorsale (a); ventrale (b)	70
5.12	Soluzione leggera: tensioni negli elementi longitudinali, vista complessiva (a) e dettaglio (c), forze (b)	71
5.13	Soluzione rigida: tensioni negli elementi longitudinali, vista complessiva (a) e dettaglio (c), forze (b)	72
5.14	Soluzione leggera: flusso delle tensioni principali nel pannello dorsale (a); dettaglio alla radice (b)	73
5.15	Soluzione leggera: flusso delle tensioni principali nel pannello ventrale (a); dettaglio alla radice (b)	74
5.16	Soluzione rigida: flusso delle tensioni principali nel pannello dorsale (a); dettaglio alla radice (b)	75
5.17	Soluzione rigida: flusso delle tensioni principali nel pannello ventrale (a); dettaglio alla radice (b)	76
5.18	Ottimizzazione del passo delle centine: peso del cassone (a); freccia massima percentuale (b)	79
5.19	Ottimizzazione del passo delle centine: passo <i>inboard</i> (a) ed <i>outboard</i> (b) .	80
5.20	Ottimizzazione del passo delle centine: distribuzione statistica del passo <i>inboard</i> (a) ed <i>outboard</i> (b)	81
5.21	Ottimizzazione del passo delle centine: visualizzazione di soluzioni appartenenti ad un certo intervallo di valori	82
5.22	Ottimizzazione dello spessore delle centine: peso del cassone (a); freccia massima percentuale (b)	84
5.23	Ottimizzazione dello spessore delle centine: <i>scatter</i> freccia-peso (a) e andamento degli spessori (b)	85
6.1	Esempio della molla non lineare	90
6.2	Primo esempio di strategia risolutiva sbagliata: materiale incrudente	91
6.3	Secondo esempio di strategia risolutiva sbagliata: ciclo iterativo intrappolato	91
6.4	Rimedio all'intrappolamento del ciclo iterativo	92
6.5	Esempio di uso del metodo della lunghezza d'arco	93
6.6	Confronto tra metodi ad incremento di carico (Newton) e di lunghezza d'arco (Crisfield) nella risoluzione del problema dell'instabilità a scatto	93
6.7	Diagramma di flusso modificato	95
6.8	Da sinistra a destra: campo di tensione tangenziale, diagonale incompleto e puro	96

6.9	Sistemi di riferiment locale e globale del laminato	98
6.10	Accoppiamenti meccanici nel laminato: taglio-estensione (a); flessione-estensione (b); flessione-torsione (c)	101
6.11	Sforzi interlaminari causati da una eccessiva differenza di coefficienti di Poisson di due laminati	102
A.1	Diagramma di flusso del progetto <i>modeFrontier</i>	112
A.2	La frontiera di Pareto	113
A.3	Schema logico dei blocchi Matlab	116
A.4	Esempi di <i>mesh</i> di un quadrato 10x10: a) GEL=2 ; b) GEL=2, <i>mesh seed</i> sul lato superiore; c) GEL=2, <i>mesh seed</i> sui lati superiore ed inferiore	119
B.1	Esempi di elementi distorti	123
B.2	Effetto cerniera lampo	123
B.3	Effetto del comando di <i>equivalence</i> : a) geometria iniziale ; b) <i>mesh</i> delle due superfici (con nodi all'interfaccia coincidenti); c) <i>Equivalence</i> con tolleranza corretta ; d) <i>Equivalence</i> con tolleranza eccessiva ed elementi collassati . . .	124
B.4	Diagramma di flusso di PROGRAMMA.m	125
B.5	Creazione della geometria esterna del cassone	126
B.6	Creazione dei piani e delle curve	128
B.7	Decomposizione del pannello <i>inboard</i>	130
B.8	Decomposizione del pannello <i>outboard</i>	131
B.9	<i>Mesh</i> delle centine	132
B.10	Modello 'sconnesso'	134
B.11	Modello vincolato	136
B.12	Modello caricato	137
B.13	Schema della numerazione degli elementi	139
B.14	Fascette di gruppi soluzione	139
C.1	Esempio di diagrammi in apertura: a) margini di sicurezza ; b) sollecitazione . . .	142
D.1	Riga di comando nell' <i>input</i> del solutore che richiama la sequenza non lineare . . .	144

ELENCO DELLE TABELLE

2.1	Principali dimensioni geometriche dell'ala	5
2.2	Principali dimensioni geometriche del cassone	6
3.1	Condizioni di volo scelte	30
3.2	Valori numerici	30
3.3	Verifiche sui carichi	37
3.4	Caratteristiche meccaniche della lega Al-2024-T3	39
4.1	Variabili in ingresso nel ciclo di ottimizzazione	50
4.2	Vincoli e obiettivi del processo	55
5.1	Variabili negli esempi di ottimizzazione	58
5.2	Parametri dell'ottimizzazione dei correnti	63
5.3	Confronto tra le soluzioni	64
5.4	Parametri dell'ottimizzazione del passo delle centine	77
5.5	Parametri dell'ottimizzazione dello spessore delle centine	83
D.1	Ingressi della <i>card</i> NLPARM	143